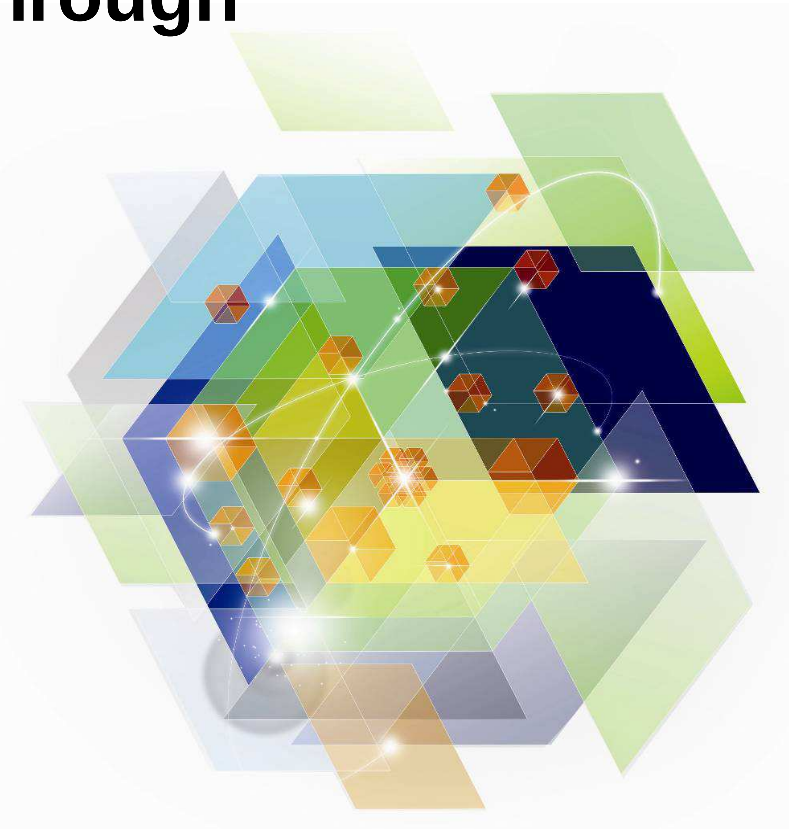
Delivering real cost savings through DB2 utilities management

TRIDUG - August, 2013

Craig Friske

Senior Software Engineer

friske@us.ibm.com



Disclaimer

– *Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision. The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.*

DB2 V10 Utilities Update

- Utility enablement of core function
- REORG
- LOAD/UNLOAD
- COPY
- RECOVER
- RUNSTATS
- CHECK
- Other
- DB2 Sort
- Summary

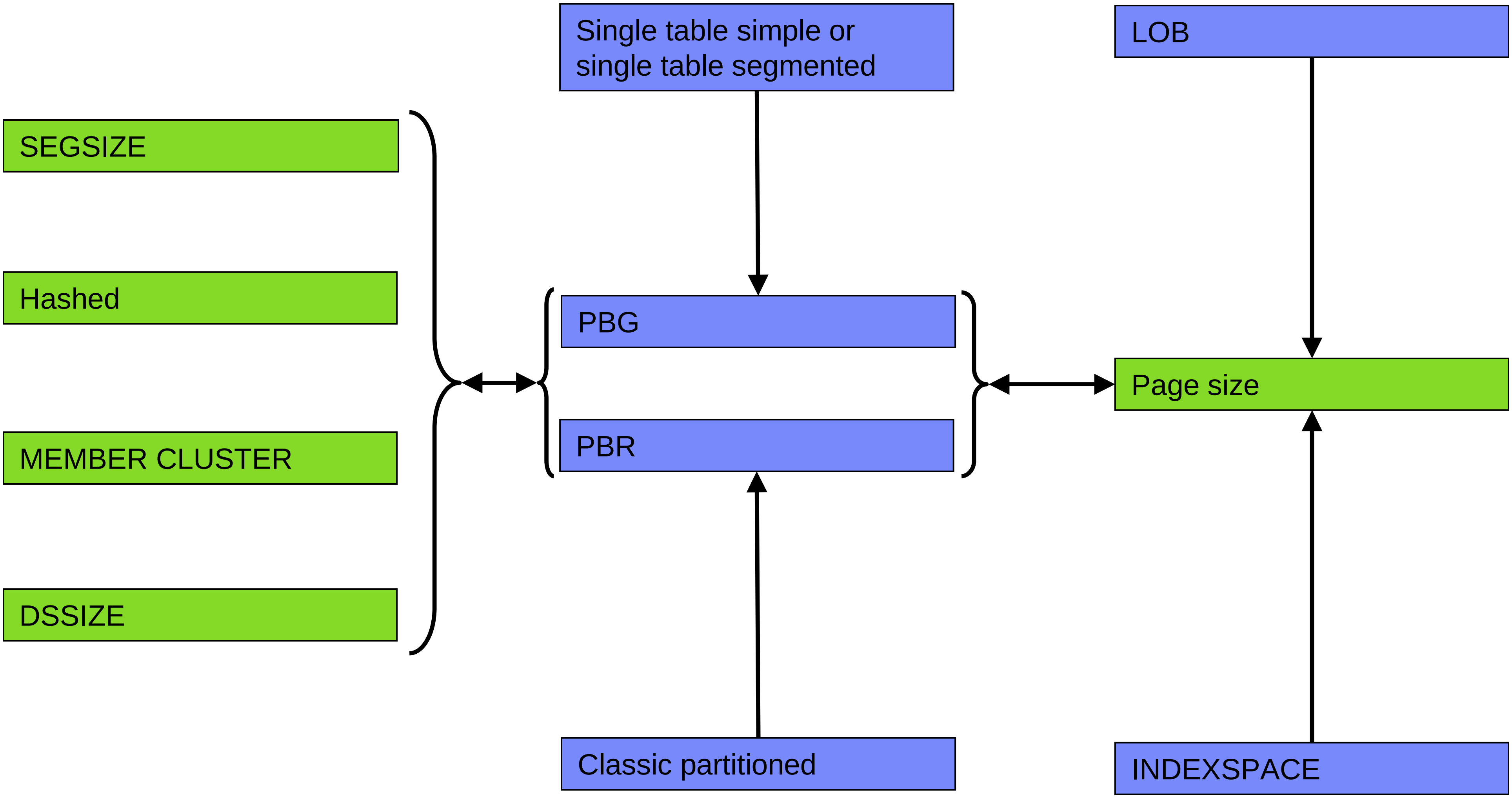
Catalog restructuring

- **What**
  - DB2 managed data sets
  - 1 table per PBG Table Space, Reordered Row Format
  - Row Level Locking
  - Eliminate links and hash chains
- **When and How (Performed by REORG during ENFM)**
  - Splitting SYSDBASE tables out into separate PBGs
  - Special REORG with inline image copies
- **Why**
  - Exploit new function, easier management
    - Automatic DS allocation
    - Reduced contention between DDL/BIND/PREPARE/Utilities
    - Eliminate 64GB limit on SPT01 with inline LOBs, compression
    - Enables REORG SHRLEVEL CHANGE catalog & directory

Online Schema

- **V9:**
  - Change of table - or index space attributes can require an outage
    - Unload data
    - Drop table space
    - Recreate table space, tables, indexes, views
    - Re-establish authorization & RI
    - Reload data
    - Undo of changes requires same process
- **V10:**
  - Execute ALTER statement (Sets AREOR state)
  - Changes are cached and materialized on next table/index space-level REORG
    - SHRLEVEL REFERENCE or CHANGE
    - LOAD REPLACE will not materialize
  - Pending changes can be dropped
  - Some restrictions exist
    - Mixing DDL
    - PIT recovery

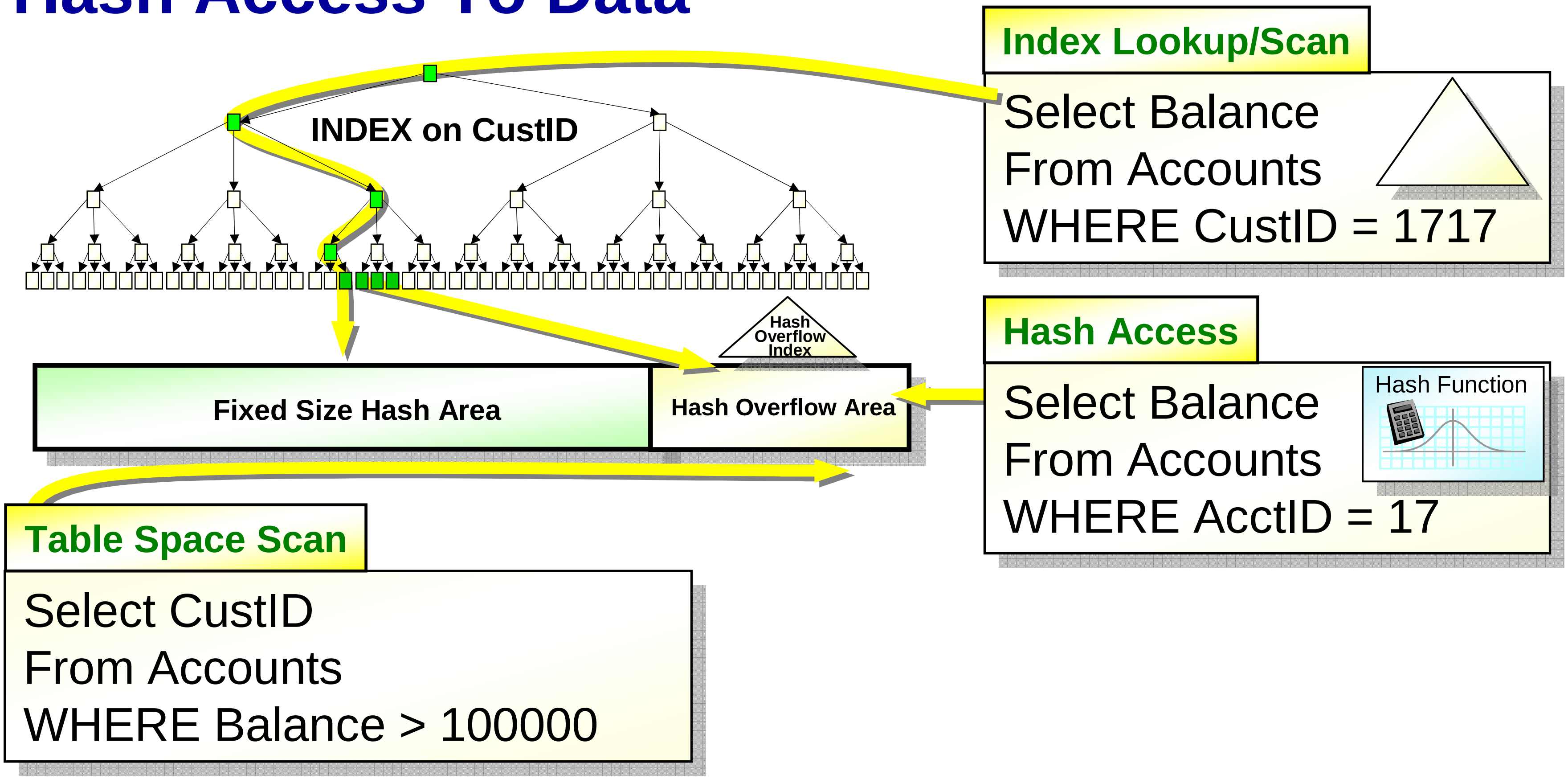
Online Schema – V10



Inline LOBs

- **Inline whole or portion of LOB with base row**
  - Universal Table Spaces (UTS) only with RRF
- **Improved performance for small LOBs**
  - Index support
- **Enables compression for inlined LOB data**
- **Immediate ALTER**
  - ALTER to increase size of inline portion sets AREO*
  - ALTER to decrease size of inline portion sets REORP
- **Requires REORG to inline existing LOB data**
  - SHRLEVEL REFERENCE or CHANGE supported

Hash Access To Data



Index Lookup/Scan

Select Balance
From Accounts
WHERE CustID = 1717

Hash Access

Select Balance
From Accounts
WHERE AcctID = 17

Table Space Scan

Select CustID
From Accounts
WHERE Balance > 100000

- **Hash Access provides the ability to directly locate a row in a table without having to use an index**
- **New AUTOESTSPACE option on REORG to manage & size hash table spaces**

REORG – improved availability & removed restrictions

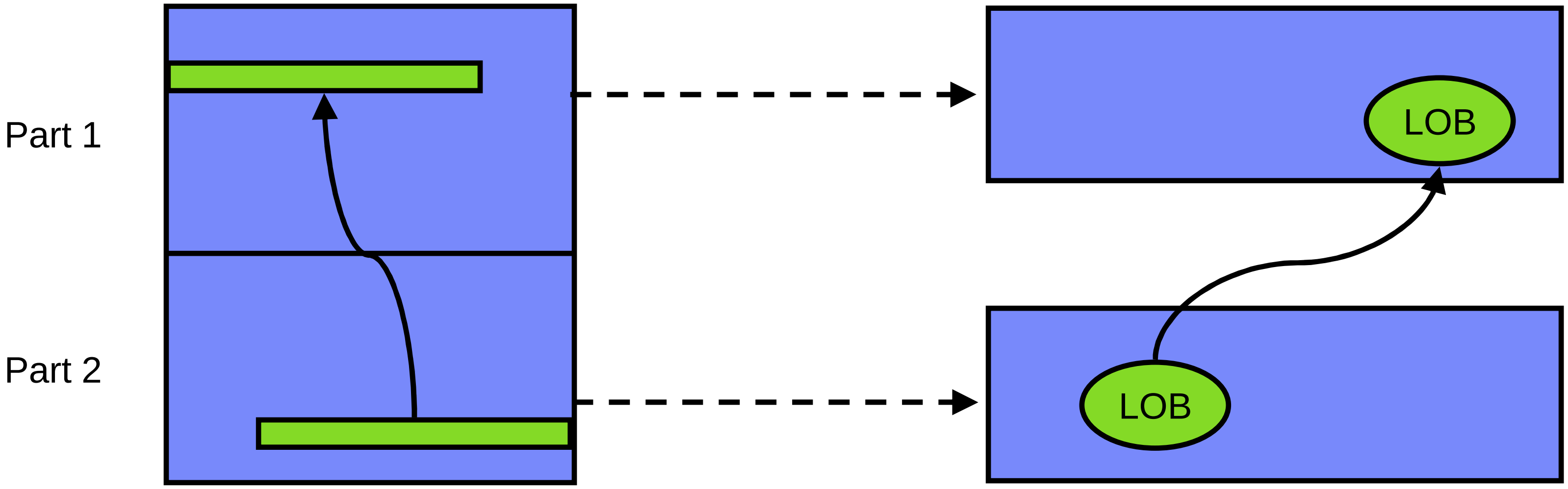
- **Reduced need for REORG INDEX**
  - List prefetch of index leaf pages based on non-leaf information for range scans
- **Improved performance for part-level REORG with NPIs & REORG INDEX**
  - Index list prefetch results in up to 60% elapsed time reduction
- **Reduced need for REORG with compress on insert**
- **New REORGCLUSTERSENS RTS column**
  - If no clustering-sensitive queries then avoid REORG to restore clustering
  - DSNACCOX enhanced
- **REORG SHRLEVEL CHANGE for all cat/dir page sets**
- **REORG SHRLEVEL REFERENCE|CHANGE to remove REORP**

REORG – improved availability & removed restrictions

- **REORG SHRLEVEL CHANGE for LOBs**
  - – Independent of whether LOBs are LOG NO or LOG YES
  - – No mapping table required
  - – Base table space must be LOGGED
- **Reduced application outage for REORG, especially with inline stats**
  - – Dedrain at beginning of UTILTERM phase, and update catalog after dedrain
- **REORG support for multiple part ranges**
  - – REORG support retrofitted to V9 in PK87762
    - LISTDEF support is not retrofitted
    - E.g. REORG PART 1,45:71,500:503,4010
    - More efficient, improved availability, exploit parallelism
- **REORG FORCE option to cancel blocking threads**
  - – FORCE ALL or just READERS
  - – Same process as –CANCEL THREAD so requires thread to be active in DB2 for it to be cancelled
  - – Threads cancelled on final drain

REORG – improved availability & removed restrictions

- **New AUX keyword on REORG of partitioned base for improved LOB handling**
  - Permit rows to flow between partitions
  - Allows REORG REBALANCE with LOB columns
  - Allows ALTER of LIMITKEY with LOB columns
  - Permits move of rows between parts on PBG REORG
  - Permits deletion of corresponding LOBs on REORG DISCARD
  - Default is AUX NO unless LOB objects required to complete REORG
  - No XML column support for classic partitioned or PBR
  - No mapping table change



Improve performance of part-level REORG with NPSIs

- **New option SORTNPSI to defer shadow index build until all keys passed through sort**
- **New parm & zparm to govern**
  - AUTO/ENABLE/DISABLE options
- **Greater % partitions and data yields better ET.**
- **Result:**
  - Customer test of REORG of 40% of partitions showed 55% ET reduction & 22% CPU increase
  - DB2 Sort gives additional ET reduction & cuts CPU to less than original starting point
- **Retrofit to DB2 9 PM55051**

New solutions for performance & CPU reduction

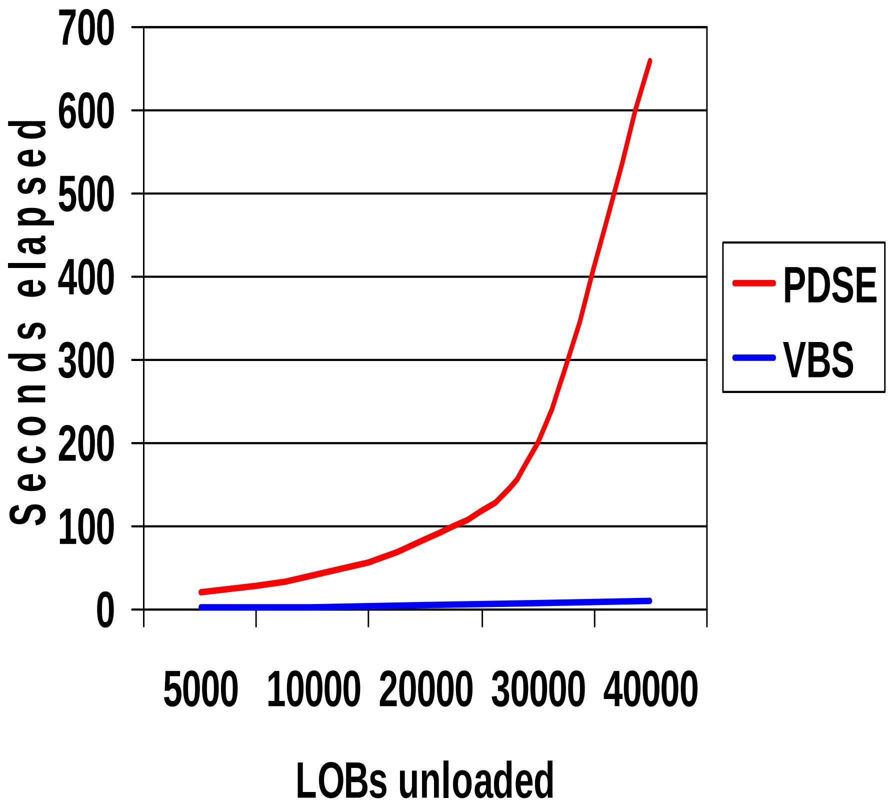
- **LOAD/UNLOAD FORMAT INTERNAL**
  - PM19584, PM35284 (V9 retrofit)
  - Unload and load data in true internal format (BRF or RRF)
  - Avoid field processing
  - UNLOAD customer measurement 80% CPU reduction, 80% ET reduction
  - LOAD measurement 77% elapsed time, 56% CPU reduction
- **LOAD PRESORTED**
  - PM19584, PM35284 (V9 retrofit)
  - Avoid sort overhead
  - Up to 25% CPU reduction, 33% ET reduction depending on # of indexes
  - Can combine with Utility Enhancement Tool PRESORT option

New solutions for performance & CPU reduction

- **LOAD INDEXDEFER**
  - LOAD bypasses the sort and build of the indexes during the LOAD phase, and leaves the deferred indexes in Rebuild Pending (RBDP) status
  - Do REBUILD IX afterwards
  - Potential savings in CPU time
  - DEFER ALL, NPIs, or NONUNIQUE
- **INSERT (instead of LOAD)**
  - Deferred index build can be done by using REPAIR to set RBDP or PSRBDP on non-unique indexes before insert processing

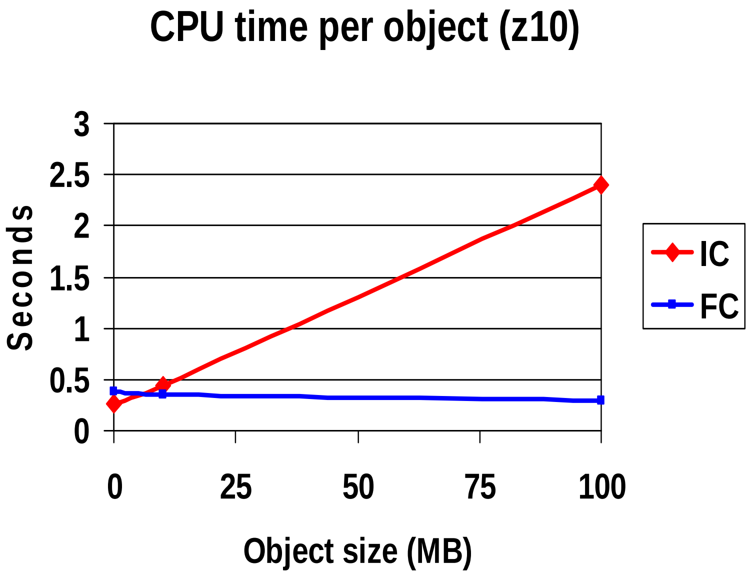
LOAD/UNLOAD

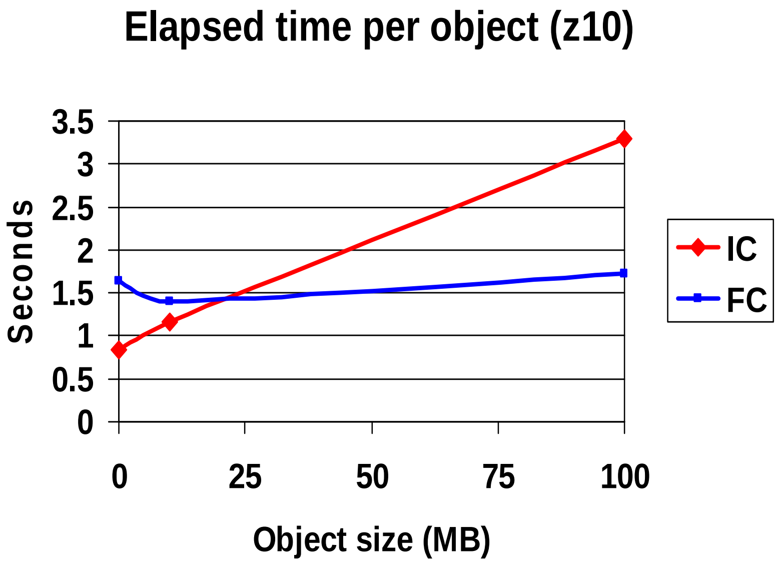
- **Remove MAX_UTIL_PARTS zparm**
  - Create unique dictionary per part for LOAD <=254 parts
  - Continue to create single dictionary if loading >254 parts
- **Improved performance for LOAD REPLACE with LOB data**
  - Up to 50% elapsed time reduction
- **Spanned record support for LOB/XML data**
  - LOBs & XML documents inlined in SYSREC with base data
  - Option in addition to FRVs
  - Performance & portability



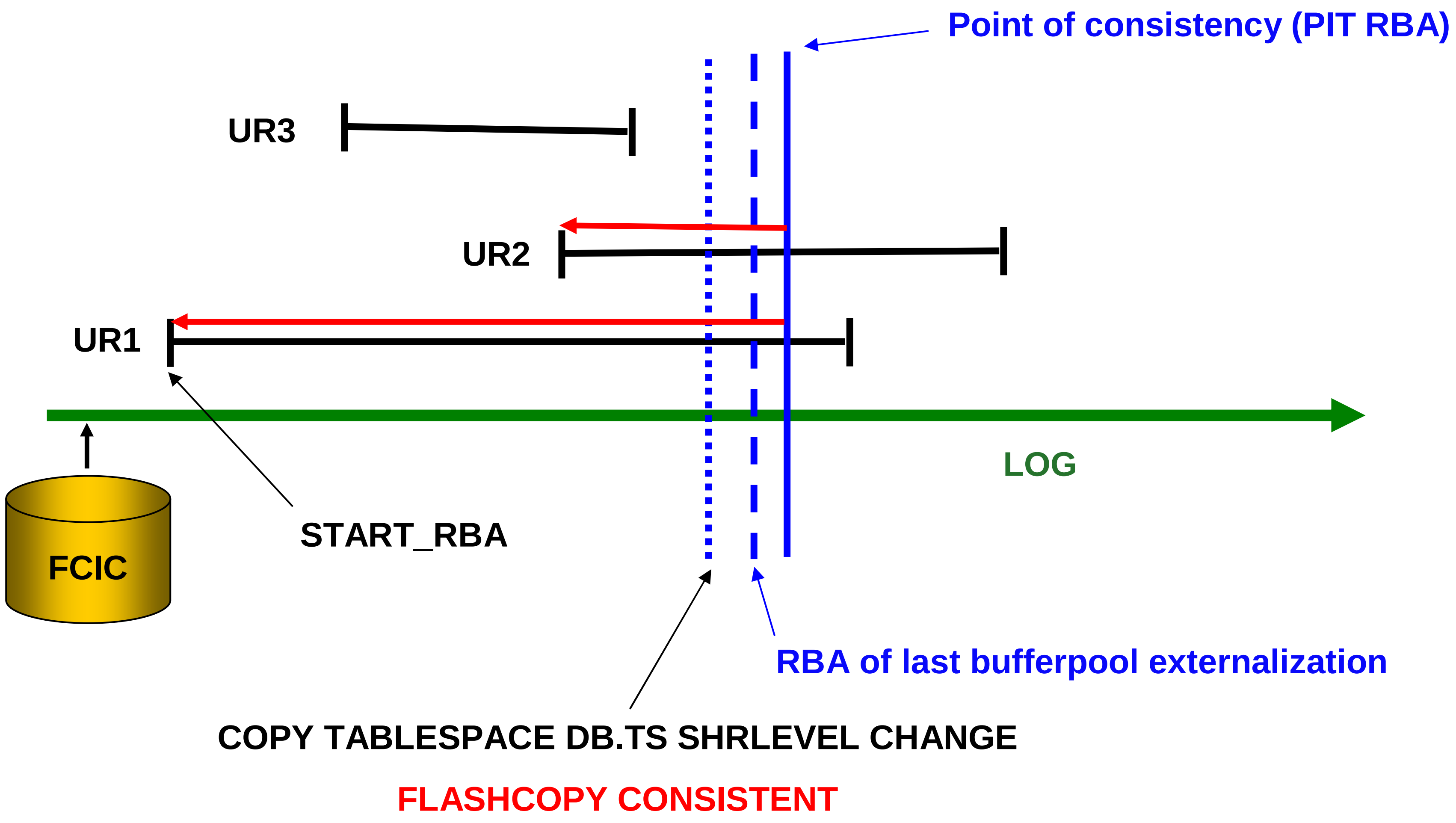
COPY

- **Dataset-level Flashcopy support**
  - COPY, RECOVER, REORG, LOAD, REBUILD INDEX, REORG INDEX
  - Significant CPU & elapsed time reduction for large page sets
  - Create transaction-consistent image copies from SHRLEVEL CHANGE
    - FLASHCOPY CONSISTENT
  - No incremental copy support





Flash Copy Image Copy (FCIC) Consistent


Point of consistency (PIT RBA)
UR3
UR2
UR1
LOG
FCIC
START_RBA
RBA of last bufferpool externalization
COPY TABLESPACE DB.TS SHRLEVEL CHANGE
FLASHCOPY CONSISTENT

Utility FlashCopy enhancements

- **Additional zparms**
  - Default settings FLASHCOPY (NO/YES) for COPY, LOAD, REORG, REBUILD
  - FAST REPLICATION: behavior with CHECK utilities (PREFERRED/REQUIRED)
  - FLASHCOPY PPRC: behavior in PPRC environment (NONE/PREFERRED/REQUIRED)
  - FAST RESTORE: behavior with BACKUP SYSTEM (NONE/PREFERRED/REQUIRED)
- **Support for REORG AUX YES to FlashCopy aux objects**
- **Remove requirement for sequential inline copy for REORG**

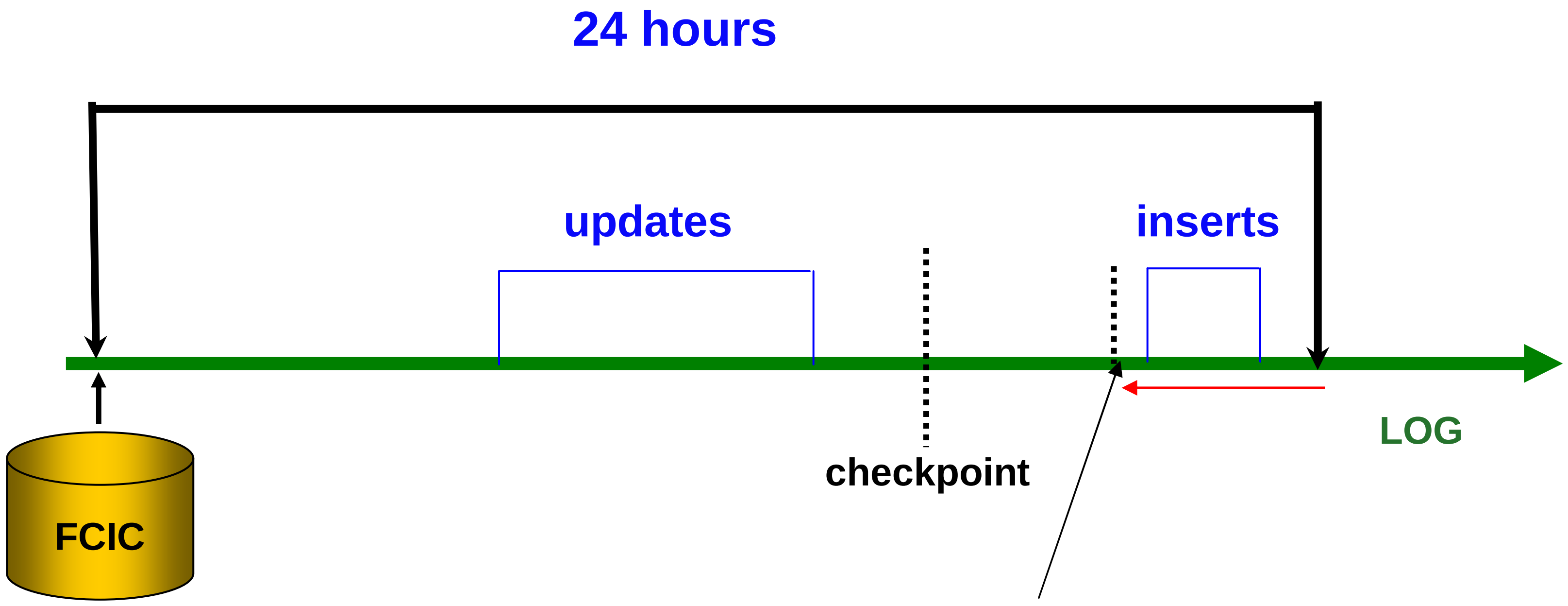
COPY

- **Improved dataset management & performance**
  - –CHANGELIMIT to use RTS to determine full vs. incremental copy
  - –Incremental copy will not allocate copy dataset unless pages changed
  - –Incremental image copy SYSUTILX logging reduction, particularly for MEMBER CLUSTER
    - Up to 70% logging rate reduction, 30-50% ET reduction, 6-14% CPU reduction

RECOVER

- **VERIFYSET option to fail PIT recovery if entire set not included**
  - – Base, LOB, XML, history – not RI
- **ENFORCE option to avoid CHKP/ACHKP when subset of set recovered**
  - – Improved performance due to avoidance of set checking (RI, aux)
- **New BACKOUT YES option for point in time recovery**
  - – True rollback, not run of generated SQL undo statements
  - – Requires COPY YES for indexes
  - – LOB tables spaces supported
  - – Restricted across mass delete

RECOVER BACKOUT YES



REOCVER TABLESPACE DB.TS TO RBA

BACKOUT YES

RUNSTATS

- **RUNSTATS PROFILE support for simplification**
- **Autonomic features through new stored procedures & catalog tables**
- **All catalog statistics columns made updatable**
- **RUNSTATS SHRLEVEL REFERENCE updates RTS**
  - TOTALROWS & TOTALENTRIES columns
- **zIIP-enablement for RUNSTATS**
  - >90% offload in some cases, not for COLGROUP
- **Auto sampling rates & page sampling instead of row sampling**
  - TABLESAMPLE SYSTEM AUTO
  - Significant CPU & ET savings
    - E.g. ET: 9:53mins to 2:30mins, CPU: 263secs to 2 secs

RUNSTATS tip: Avoid redundant index stats collection

- Beware of using LISTDEF for parallelism to reduce elapsed time

```
LISTDEF CPLST1 INCLUDE TABLESPACE NFMBIGLT.NLT1TS1 PARTLEVEL(1:333)
LISTDEF CPLST1 INCLUDE TABLESPACE NFMBIGLT.NLT1TS1 PARTLEVEL(334:666)
LISTDEF CPLST1 INCLUDE TABLESPACE NFMBIGLT.NLT1TS1 PARTLEVEL(1000:1333)
LISTDEF CPLST1 INCLUDE TABLESPACE NFMBIGLT.NLT1TS1 PARTLEVEL(1334:1666)
LISTDEF CPLST6 INCLUDE TABLESPACE NFMBIGLT.NLT1TS1 PARTLEVEL(1677:2048)
RUNSTATS TABLESPACE LIST CPLST1 INDEX(ALL) TABLE(ALL) UPDATE ALL
```

VS

```
LISTDEF CPLST1 INCLUDE TABLESPACE NFMBIGLT.NLT1TS1
RUNSTATS TABLESPACE LIST CPLST1 INDEX(ALL) TABLE(ALL) UPDATE ALL
```

- Over 2 days elapsed time vs 3 hours elapsed time

CHECK

- **CHECK utilities will no longer set CHKP/ACHKP**
- **CHECK SHRLEVEL CHANGE default changed to fail if Flashcopy not available**
  - Zparm to govern
  - New UTIL_TEMP_STORCLAS zparm
  - Prevent CHECK utilities from impacting XRC/PPRC/GDPS & BACKUP SYSTEM
- **CHECK DATA enhanced for XML support**
  - Document validation
  - Schema validation
- **Automated exception table processing for XML documents**

Other

- **Removed UTSERIAL lock for greater utility concurrency**

- **LISTDEF & TEMPLATE enhancements**
  - LISTDEF support for multiple part ranges on REORG
  - LISTDEF support for DEFINED YES|NO|ALL
    - Improved utility performance since unnecessary to build & then discard structures for undefined objects
    - Default changed to DEFINED YES & empty lists result in RC4 (not RC8)

- **REPORT RECOVERY support for SLBs**

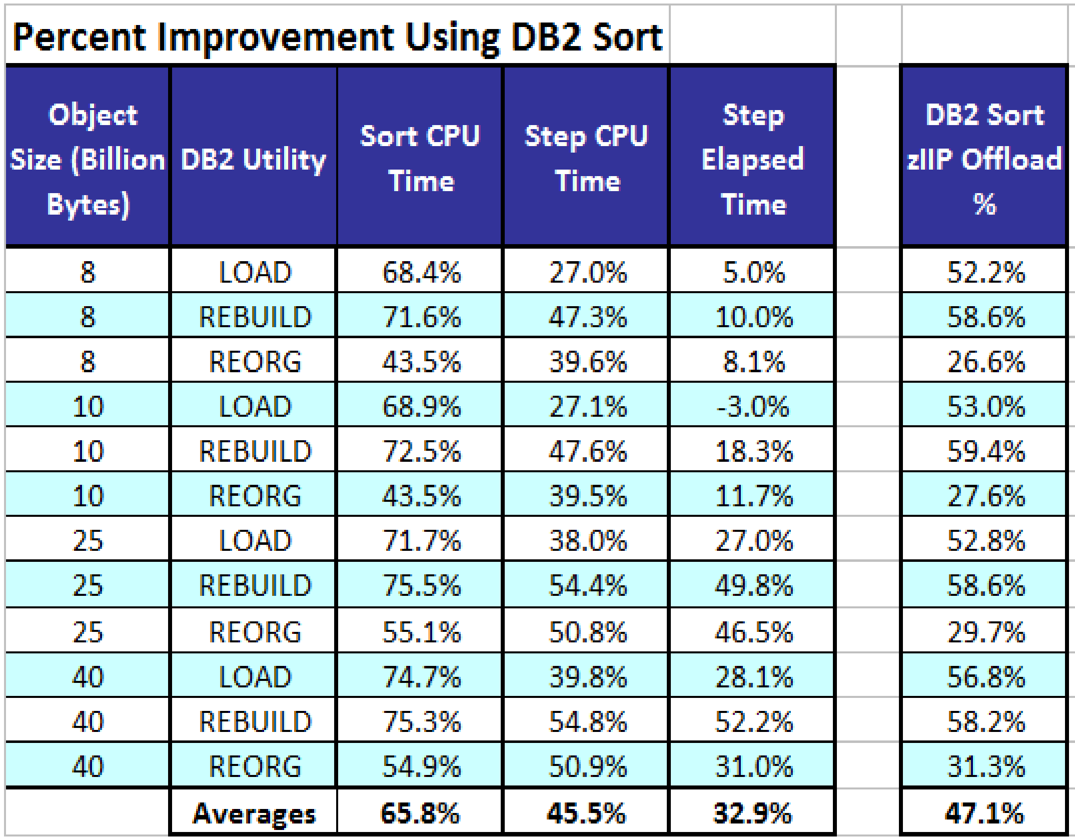
- **SQL SELECT on SYSLGRNX**

V10 with DB2 Sort 1.3 Performance Benefits

Use of DB2 Sort 1.3 with DB2 utilities, as compared with running DB2 utilities alone, may see: *

- Reduction of Sort CPU usage
  - Up to 74% reduction on machines with zIIP engines
  - Up to 43% reduction on machines without zIIP engines
- Reduction of Utility CPU usage
  - Up to 49% reduction on machines with zIIP engines
  - Up to 25% reduction on machines without zIIP engines
- Reduction of Utility Elapsed Time
  - Up to 50% reduction on machines with zIIP engines
  - Up to 49% reduction on machines without zIIP engines

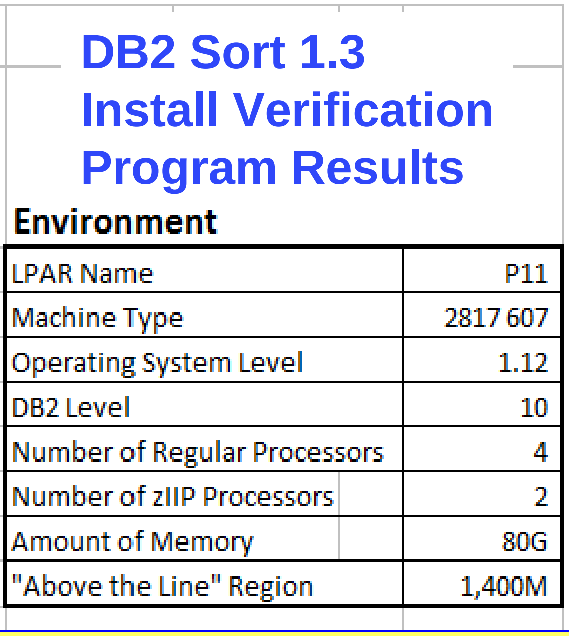
DB2 Sort 1.3 Beta - Customer Results



Percent Improvement Using DB2 Sort

Object Size (Billion Bytes)	DB2 Utility	Sort CPU Time	Step CPU Time	Step Elapsed Time	DB2 Sort zIIP Offload %
8	LOAD	68.4%	27.0%	5.0%	52.2%
8	REBUILD	71.6%	47.3%	10.0%	58.6%
8	REORG	43.5%	39.6%	8.1%	26.6%
10	LOAD	68.9%	27.1%	-3.0%	53.0%
10	REBUILD	72.5%	47.6%	18.3%	59.4%
10	REORG	43.5%	39.5%	11.7%	27.6%
25	LOAD	71.7%	38.0%	27.0%	52.8%
25	REBUILD	75.5%	54.4%	49.8%	58.6%
25	REORG	55.1%	50.8%	46.5%	29.7%
40	LOAD	74.7%	39.8%	28.1%	56.8%
40	REBUILD	75.3%	54.8%	52.2%	58.2%
40	REORG	54.9%	50.9%	31.0%	31.3%
	Averages	**65.8%**	**45.5%**	**32.9%**	**47.1%**

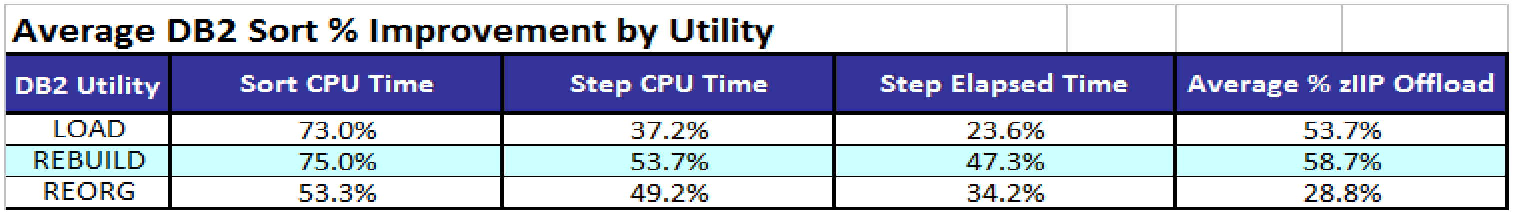
DB2 Sort 1.3 Install Verification Program Results



Environment

LPAR Name	P11
Machine Type	2817 607
Operating System Level	1.12
DB2 Level	10
Number of Regular Processors	4
Number of zIIP Processors	2
Amount of Memory	80G
"Above the Line" Region	1,400M

Webcast: DB2 Sort – A Customer Experience and Technical Details Replay available at: http://goo.gl/bzngn



Average DB2 Sort % Improvement by Utility

DB2 Utility	Sort CPU Time	Step CPU Time	Step Elapsed Time	Average % zIIP Offload
LOAD	73.0%	37.2%	23.6%	53.7%
REBUILD	75.0%	53.7%	47.3%	58.7%
REORG	53.3%	49.2%	34.2%	28.8%

Based on system reports of CPU utilization and elapsed time generated in the specific customer's environment and provided to IBM. Results obtained in other operating environments may differ significantly.

DB2 utilities future priorities **

- REORG
  - Switch Phase Relief and performance
  - Better PBG and mapping table management
  - REBALANCE availability and reliability
- RUNSTATS
  - zIIP offload,
  - inline stats
- RECOVER
  - RECOVER PIT prior to schema changes, cloning help
- LOAD/UNLOAD extended parallelism and fetch continue
- Parallelism controls
- REPAIR catalog

***** The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality.***

Summary

- **Superior DB2 for z/OS support from beta, GA and beyond**
- **Over the last several releases IBM’s has had a focus on performance, performance and performance –**
  - –Reducing cost of ownership with CPU, ET and zIIP offload

Questions?

Thank
You