



(<https://www.idug.org/>)

Login (<https://www.idug.org/l/li/in/>) | Request Login ([l/pw/rs/](https://www.idug.org/l/pw/rs/))

(<https://www.idug.org/emea2019> )



IDUG Content Blog ([p/bl/et/blogid=278](https://www.idug.org/p/bl/et/blogid=278))

## PBR RPN explained



**Haakon Roberts**

## PBR RPN explained.

Haakon Roberts, DB2 for z/OS Development

### Introduction.

DB2 tables continue to grow in size and these days it is not unusual to encounter tables containing tens of billions of rows. Consequently, DB2 for z/OS must continue to evolve to ensure such large amounts of data can be managed in an efficient manner with minimal or no application impact.

A significant step forward in this regard is the introduction in DB2 12 for z/OS of a major new feature called “PBR RPN” which transforms the management of range-partitioned table spaces. But what really is PBR RPN and why should it matter to me? This article explains the reasons behind this development effort, the issues RPN addresses, the benefits and use cases and also gives advice on its exploitation.

## Why the Need to Change?

Range-partitioned table spaces typically house the largest tables in a DB2 environment. As these tables continue to grow in size, the efficient management of them becomes increasingly important. This means avoiding where possible the need to process the entire table space in one go in REORG. Reorging individual partitions is fine unless a schema change is involved. The concept of deferred ALTERs followed by materializing REORGs was introduced in V10, and, since the attributes being changed are all at table space-level, that means that a table space-level REORG is required to effect the change. Such REORGs are becoming increasingly unpalatable as the size of these tables continues to grow.

A second issue is that there are ever-more stringent requirements for application and data availability. The concern is that if one runs out of space in a partition of a range-partitioned table space then application outages will occur since DB2 does not have the option to place new rows in other partitions due to the requirement to adhere to the partitioning scheme.

Running out of space in a partition can be addressed by either altering the limit key and rebalancing data across partitions through REORG, or using the REORG utility REBALANCE option to allow the REORG utility to choose an optimal limit key value to rebalance data across multiple partitions. Both options are available in V11 and can be run with minimal application impact through online REORG.

Even so, this is not enough. For many customers applications are dependent upon the partitioning scheme, making changing the limit key values onerous. This makes it more desirable to be able to increase the partition size (DSSIZE) for partitions. This is possible in V11 through deferred ALTER and materializing REORG, but there are a couple of problems:

1. There is a complex relationship between DSSIZE, number of partitions and page size meaning that it might not be possible to increase DSSIZE. This relationship is difficult to understand and manage. The same problem can also prevent customers from being able to add partitions.
2. Even if DSSIZE can be increased, it's a table space-level attribute which requires a REORG of the entire table space for it to take effect, and it affects all partitions so all partitions increase in size.

The underlying reason for these issues is the format of the 5 byte RID in DB2. The RID uniquely identifies each row in a table space. For partitioned table spaces, the 4 byte page number includes a number of bits that identify the partition number. So the more partitions the fewer bits are left for the page number and vice versa. We say that the page number is "absolute" because it identifies the page number and partition within the table space.

## Introducing PBR RPN.

The solution to these issues is two-fold.

First of all, DB2 has separated the partition number from the page number. This means that now RIDs are 7 bytes instead of 5: A 2-byte partition number, a 4-byte page number and a 1-byte IDMAP value (as before). Since the page number no longer contains the partition number, the page number is relative to the start of a given partition. Hence this enhancement is called "Relative Page Numbering", i.e. PBR RPN. The partition number doesn't need to be stored in each page, so one can no longer look at a data page and know which partition it belongs to. Note that this option applies to PBR table spaces only, not PBG or classic partitioned.

This means that now DB2 has broken the complex relationship between number of partitions, page size and DSSIZE. As a result, DB2 introduces a simple rule that individual partitions can now be up to 1Tb in size, regardless of number of partitions page size. Multiplied by 4096, this means that the maximum size of a PBR RPN table is 4Pb, and with a 4Kb page size and up to 255 rows per page, it equates to roughly 280 trillion rows.

The second piece of the PBR RPN solution is to start to drive some attributes from table space level down to partition level. In DB2 12 the primary focus is on DSSIZE. DSSIZE is now stored in SYSTABLEPART in addition to SYSTABLESPACE. An ALTER to increase DSSIZE for a PBR RPN table space is now possible for a single partition. Not only that, it is an immediate ALTER not even requiring a REORG to materialize. This represents a massive availability and usability improvement for PBR table spaces.

If you have LOB or XML columns in a table that is in a PBR RPN table space, then the XML table space will automatically be PBR RPN too and benefit from the same attributes. LOB table spaces are unaffected by PBR RPN.

Similarly, partitioned indexes (PI and DPSI) on PBR RPN table spaces now support DSSIZE on CREATE INDEX and ALTER INDEX, allowing you to manage index partition sizes independently of table space size.

## The Benefits Defined

What does PBR RPN mean for you in practical terms? Here's a concise list explaining why this new feature in V12 is so important.

- The maximum size of a range partitioned table space and therefore the maximum size of a given table in DB2 increases to 4Pb, containing up to 280Tn rows.
- DSSIZE can now be increased up to 1Tb through an immediate ALTER at the partition level. No application impact, no need for a REORG at all and complex considerations regarding page size or number of partitions.
- Index partition sizes can now be up to 1Tb in size and managed independently of table space DSSIZE.
- Log records can now be processed by partition, simplifying DSN1LOGP processing.

## Conversion and Exploitation

RPN is an attribute of PBR table spaces, not a new table space type. It's available in DB2 12 from M500 onwards. There are two options: Create new or convert existing table spaces.

DB2 12 introduces new zparm PAGESET\_PAGENUM. The options are ABSOLUTE or RELATIVE.

CREATE TABLESPACE has also been enhanced to support a PAGENUM attribute where the options are ABSOLUTE or RELATIVE. The CREATE TABLESPACE option, if specified, overrides the zparm.

Conversion of existing range-partitioned table spaces is performed through ALTER TABLESPACE... PAGENUM RELATIVE. The table space will be placed in AREOR state and conversion is then performed through online REORG. It requires a REORG of the entire tablespace. Both classic partitioned and PBR APN table spaces can be converted. Note that there is no support for altering back to ABSOLUTE, just as there is no support for converting back from PBR to "classic" partitioned.

## Usage considerations

There are a number of factors to consider in V12, some of which apply whether you choose to exploit PBR RPN or not.

- There is a new log record format to support 7 byte RIDs. This log record format is written from M100 onwards, which means that it's critical that fallback toleration code is applied correctly and that any product that reads DB2 log records has the ability to support the new format log records. Note that this log record format is used in V12 regardless of whether you plan to exploit PBR RPN or not.
- There is a new mapping table format for REORG SHRLEVEL CHANGE. Support for 7 byte RIDs follows the same method as for 10 byte log RBAs in V11. This means that in V12 M100 online REORG will support both V11 and V12 format mapping tables. Once function level M500 or above is activated, the REORG utility will only accept V12 format mapping tables. If a V11 format mapping table is given (or no mapping table at all), then the REORG utility will ignore it, create its own and drop it at the end of the REORG or at the end of a sequence of REORGs. Note that the new mapping table format is used irrespective of whether you are using PBR RPN or not.
- The conversion to PBR RPN requires a REORG of the entire table space.
- Just as there is no reversion from PBR to classic partitioned, there is no reversion from PBR RPN to "APN".
- There is no support for recovering to a point in time prior to a REORG that converts to PBR RPN.
- REORG of PBR RPN or to convert to PBR RPN currently requires partition-level inline image copies. This may be an issue if inline image copies are written to tape as the number of concurrent tape units consumed is likely to exceed the number available. Operational relief is possible by writing inline image copies to disk, however IBM is planning a solution for this issue to be delivered via an APAR in DB2 12. No APAR number exists at this time.
- Some range partitioned table spaces can have limit key values that are internally truncated to 40 bytes. This can occur if the table space was created prior to V6 and the limitkey defined exceeded the 40 byte maximum at the time. This internal truncation is not supported by PBR RPN. Conversion of such a table space to RPN will fail. IBM intends to provide a solution to remove this truncation thereby allowing conversion of such table spaces to RPN. No date is finalized yet for the delivery of such a solution.

## Summary

PBR RPN represents a major improvement in terms of scalability, usability and availability for managing large, range-partitioned table spaces with a far simpler set of rules. Moreover, it creates a firm foundation for the delivery of future enhancements for range-partitioned table spaces. We firmly believe that you will see significant value from PBR RPN and are confident that you will benefit from this major feature of DB2 12.

---

**Released:** July 30, 2017 04:30 AM | **Updated:** July 30, 2017 04:30 AM

**Keywords:** DB2 for z/OS (p/bl/kw/kt=1&kw=DB2%20for%20z%20FOS) | administration (p/bl/kw/kt=2&kw=administration) | New Release DB2 (p/bl/kw/kt=2&kw=New%20Release%20DB2) | Partitioning (p/bl/kw/kt=2&kw=Partitioning)

## Recent Stories

### Db2 for z/OS – New Encryption Capabilities to Protect Your Data (p/bl/ar/blogaid=910)

Updated 6:04AM CDT, Wed Sep 11th, 2019



(p/bl/ar/blogaid=910) With many high-profile data security breaches, data encryption has become a focus item up to the CIO and CEO levels. Any loss or exposure of sensitive data might cause irreparable damage to the busine...

### Transparent Data Encryption (p/bl/ar/blogaid=902)

Updated 6:04AM CDT, Wed Sep 4th, 2019



(p/bl/ar/blogaid=902) There have been several high visibility data exposures in the news over the last year or so. As a Db2 DBA, one of your responsibilities is the protection of the data your business needs to run. Your a...

### August Content Recap (p/bl/ar/blogaid=907)

Updated 10:58AM CDT, Tue Sep 3rd, 2019



(p/bl/ar/blogaid=907) Recap of IDUG content for the month of August 2019.

### International Db2 Users Group

2601 Iron Gate Drive, #101

Wilmington, NC 28412 ([https://maps.google.com/?](https://maps.google.com/?q=2601+Iron+Gate+Drive,+%23101+Wilmington,+NC+28412&entry=gmail&source=g)

[q=2601+Iron+Gate+Drive,+%23101+Wilmington,+NC+28412&entry=gmail&source=g](https://maps.google.com/?q=2601+Iron+Gate+Drive,+%23101+Wilmington,+NC+28412&entry=gmail&source=g))

Phone: (910) 660-8649 (tel:+19106608649) / Fax: (910) 523-5504 (tel:+19105235504)

Copyright © 2019 IDUG. All Rights Reserved.

All material , files, logos, and trademarks within this site are properties of their respective organizations.



(<https://www.facebook.com/IDUGDB2>)



(<https://twitter.com/IDUGDB2>)



(<https://www.linkedin.com/grp/home?gid=46747>)



(<https://www.youtube.com/user/IDUGDB2>)