

Point-in-time recovery using DB2 9 for z/OS RESTORE SYSTEM utility and SET LOG SUSPEND

System point-in-time-recovery (PITR) can be a very complex and long running process. With the DB2 9 for z/OS RESTORE SYSTEM utility along with the right volume level copy strategy, a system PITR can be accomplished in a couple of hours, instead of days. It is not necessary to use the BACKUP SYSTEM utility in obtaining the system level copy in order to leverage the benefits of a total system level recovery using the RESTORE SYSTEM utility. The system level copy can be obtained using the volume level copy technology of your DASD along with a SET LOG SUSPEND. Please reference *Automated Online System Copy Process: DB2 Suspended Log using FlashCopy* TechDocs: <http://w3-1.ibm.com/support/techdocs/atsmastr.nsf/PubAllNum/TD101860> for help on creating an online system copy process.

RESTORE SYSTEM utility without BACKUP SYSTEM utility

The RESTORE SYSTEM utility can be used to apply logs to the database volumes that have been restored outside of DB2. The LOGONLY option of the RESTORE SYSTEM utility is used to skip the restore phase and only apply the logs. This facilitates the use of the RESTORE SYSTEM utility without having to use the BACKUP SYSTEM utility to obtain the system copy. This is helpful when the requirements for the BACKUP SYSTEM utility have not been met, making that utility unavailable for use. Requirements for BACKUP SYSTEM utility are:

- z/OS 1.5 or higher
- DASD hardware that writes to the DFSMSHsm API

In order to use the RESTORE SYSTEM utility there must be a Recovery Base Log Point (RBLP) recorded in DBD01. When not using the BACKUP SYSTEM utility the only other way to create a RBLP is through the use of the SET LOG SUSPEND command. Please reference *Automated Online System Copy Process: DB2 Suspended Log using FlashCopy* TechDocs: <http://w3-1.ibm.com/support/techdocs/atsmastr.nsf/PubAllNum/TD101860> for help on using SET LOG SUSPEND.

Configuration Planning

The DB2 system logging configuration and frequency of volume level copies are very important in helping to achieve reduced recovery time.

ICF catalogs

It is critical that the ICF catalog for the BSDSs, active logs, and archive logs is separate from the ICF catalog(s) for the DB2 catalog and directory, application VSAM datasets (tablespaces and indexes), and any program products. Also, the ICF catalog for the BSDS, active logs, and archive logs should be on either a volume with those datasets, or a volume that is separate from the DB2 catalog and directory, application VSAM datasets, and the program products.

DB2 system active and archive logs

There should be some planning given to how much time should be represented by log records on the active logs in conjunction with frequency of volume level copies. In other words, if you have a highly active system you want to think about obtaining volume level copies multiple times a day, and configure your active log environment to hold enough logs records to cover the time frame between volume level copies. So, if you obtain volume level copies three times a day (every eight hours) configure the active log environment to hold eight hours or more worth of log records.

If it is necessary to go back on archives during a PITR, give thought to how you are archiving and how quickly DB2 can gain access to the archives. If you are archiving to an SMS pool, think about how many hours you want to keep in the archive pool. If you are migrating with HSM, think about how quickly you can recall those archive datasets.

DB2 system checkpoint

DB2 system checkpoints should be based on the CHKFREQ ZPARM versus using LOGLOAD. With CHKFREQ DB2, system checkpoints are based on an established elapsed time in terms of minutes versus LOGLOAD, which is based on number of log records written. With the latter, DB2 system checkpoints can be either very close together, in terms of time, or very far apart. With CHKFREQ, the DB2 system checkpoints are consistent in time elapsed between checkpoints. Also, with CHKFREQ combined with DLDFREQ, the HPGRBRBA in updated pagesets is being updated on a regular basis making log only recovery possible.

There used to be a challenge in trying to pick the right value for CHKFREQ. It is desirable to keep the number of minutes between checkpoints small enough in order to facilitate a quicker DB2 system restart, in the event of a system crash, while at the same time keep enough DB2 system checkpoints on the checkpoint queue in the BSDS to facilitate a PITR with the current BSDS. In the past, if the point in time that the system needed to return to did not have a checkpoint on the checkpoint queue, there were time consuming steps required to provide the checkpoint information. This has been remedied with APAR PQ93548. At system restart, DB2 will start reading the log backwards from the log truncation point until a complete checkpoint is found. The message DSNR054I RESTART...BEGIN SCAN FOR CHECKPOINT will be written out indicating this is taking place.

Establishing the recover to point

Once it has been determined the DB2 system needs to go back to a point-in-time in the past, it is necessary to figure out what that time is and how it will be represented to DB2. With DB2 9 the option of representing the point-in-time to which the system needs to return can be represented in the format of date and time. SYSPITRT is used when providing a date and time of day as the log truncation point. The time is based on GMT and not local time. If having to perform a Federated Recovery across multiple DB2 data stores, the same SYSPITRT value would be used in all DB2 subsystems and members. Please reference the DB2 Utility Guide for the utility DSNJU003.

Timestamp format: CRESTART CREATE,SYSPITRT=YYYYJJJHHMMSS

For example, if it was determined the system needed to go back to the local time of 10:48:00 on June 27, 2008, and the system is 2 hours ahead of GMT, the Julian date would be 2008179 and the GMT time would be 08:48:00.

CRESTART CREATE,SYSPITRT=2008179084800

If the DB2 system to be restored is a data sharing system, then the log truncation point must be established in each member's BSDSs.

Restore using volume level copy

There should be a procedure in place to identify what volume level copy version will be used in restoring the data back to a time prior to the time in which the system must return. If multiple versions of the volume level copies are not maintained on DASD, and the system must go back to a copy older than the current volume level copy, then it will be necessary to identify the tape copies and restore those back to DASD. Again, the only DASD volumes that should be restored from the volume level copy are the volumes containing the DB2 catalog and directory, the application VSAM datasets (tablespaces and indexes), and program products. The appropriate ICF catalogs should also be included in the volumes being restored to

a prior point-in-time. The BSDSs, active logs and archive logs should remain as they were at the time of shutting DB2 down in order to perform the PITR.

Identify log truncation RBA or LRSN

Once you know the time of day to which the DB2 system must return, and you are not using the time stamp version of the system point-in-time recovery conditional restart it is necessary to find the appropriate RBA, or in the case of DB2 Data Sharing, the LRSN that will be used in establishing log truncation. With DB2 data sharing, all members must use the same LRSN when establishing log truncation.

One way to find an appropriate log truncation point is by print off the BSDS and reviewing the checkpoint queue to see if there is an appropriate time close to the application time to which you must return. If this is DB2 Data Sharing, this must be done for all members and the oldest, or smallest LRSN, must be used when establishing log truncation for all members.

If there is no longer a checkpoint on the checkpoint queue, then using the BSDS print map, identify the active or archive log that will contain the time frame to which the application must return, and use DSN1LOGP to find the RBA or LRSN representing the time log truncation point. One other method to establishing a recorded RBA or LRSN for use in a PITR is by using the QUIESCE utility for a dummy tablespace. By running the QUIESCE utility for a tablespace that is never used in an application, there will never be a contention problem in obtaining the quiesce. If the quiesce is run on a regular interval (ie: every ten minutes) a quiesce point will be recorded in SYSCOPY for the dummy tablespace. This can be used in identifying a RBA or LRSN representing a specific point-in-time that can be used as a log truncation point.

Create conditional restart

Using DSNJU003 (Change Inventory Utility) create a system recovery conditional restart card (SYSPITR or SYSPITRT) in the BSDS(s).

```
//BSDS01 EXEC PGM=DSNJU003
//STEPLIB DD DISP=SHR,DSN=DB2V8.SDSNLOAD
//SYSUT1 DD DISP=OLD,DSN=DB2V8.BSDS01
//SYSUT2 DD DISP=OLD,DSN=DB2V8.BSDS02
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSIN DD *
  CRESTART CREATE,SYSPITR=00009E378A42
  OR
  CRESTART CREATE,SYSPITRT=2008179084800
```

If this is DB2 Data Sharing then DSNJU003 must be run for all members using the same LRSN for the SYSPITR value or the same time stamp for the SYSPITRT value.

Print the BSDS to review the conditional restart control card. You will see the following:

SYSPITR SYSTEM LEVEL RECOVERY MODE RESTART

Restart DB2

When DB2 is restarted ACCESS MAINT and DEFER ALL are implied, so it is not necessary to change the ZPARM or specify ACCESS(MAINT) during restart. However, if this is a DB2 Data Sharing system then it is still necessary to deallocate all of the coupling facility structures for this data sharing groups prior to the restart. Please reference the section titled *Deallocating structures by force* in DB2 9 Data Sharing: Planning and Administration; SC18-7417-00 for information on how to deallocate the coupling facility structures.

Also, it will be necessary to unallocate the ICF catalogs in the Master address space of each z/OS LPAR using the following command:

© Copyright IBM Corporation, 2011

F CATALOG,UNALLOCATE(SYS1.UCATname)

As with any conditional restart, there will be an outstanding reply that must be responded to for each DB2 subsystem being restarted, in order for the restart of the subsystem(s) to continue.

The following messages will be written out in the system log.

```
DSNR050I -P870 DSNRRPRC DB2 STARTED IN SYSTEM RECOVER PENDING MODE
DSNR002I -P870 RESTART COMPLETED
DSNY014I -P870 DSNYSTRT DB2 WAS STARTED WITH ACCESS(MAINT)
```

Once the restart of the DB2 subsystem(s) has completed, all but one DB2 data sharing member can be stopped.

RESTORE SYSTEM utility

The RESTORE SYSTEM utility only needs one DB2 subsystem active in order to run. If it terminates and must be restarted, it must be restarted on the same DB2 data sharing member on which it was originally running. It is important that the control card specify LOGONLY.

```
//STEP1 EXEC DSNUPROC,TIME=1440,
// UTPROC='',
// SYSTEM='DSN'
//SYSIN DD *
  RESTORE SYSTEM LOGONLY
/*
```

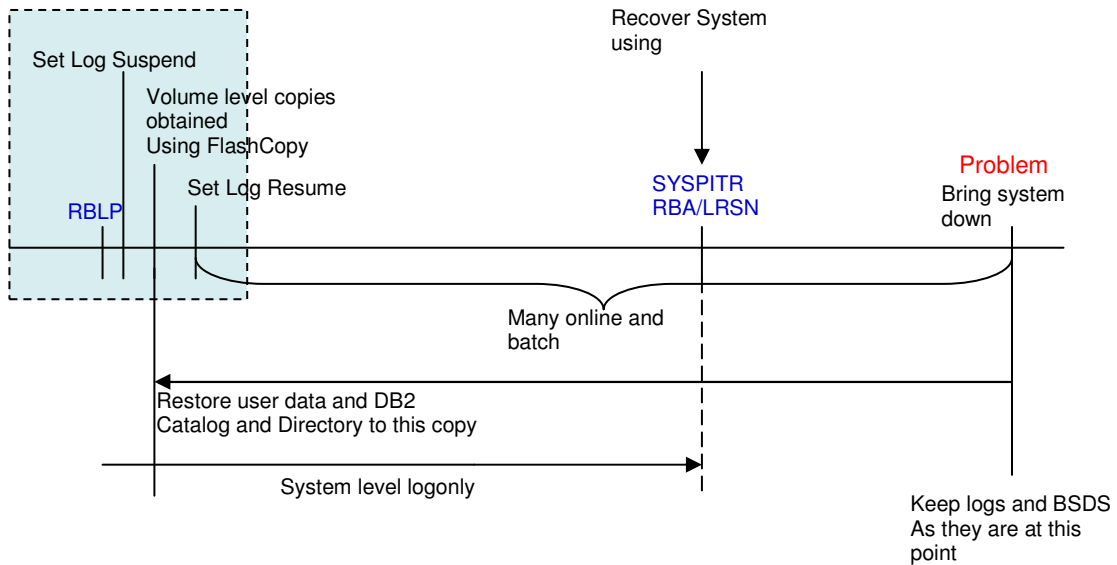
Once the RESTORE SYSTEM utility completes successfully there are a number of steps that must be completed to ensure the system is completely recovered and ready for end-user access.

- Stop and start each DB2 subsystem or member to remove it from access maintenance mode.
- Use the DISPLAY UTIL command to see if any utilities are running. If other utilities are running, use the TERM UTIL command to end them.
- Use the RECOVER utility to recover all objects in RECOVER-pending (RECP) or REBUILD-pending (RBDP) status, or use the REBUILD INDEX utility to rebuild objects.
- If a CREATE TABLESPACE, CREATE INDEX, or data set extension has failed, you can also recover or rebuild any objects in the logical page list (LPL).

Please reference the Redbook entitled *Disaster Recovery with DB2 UDB for z/OS; SG24-6370-00; Chapter 19 Local recovery: System PITR* for more detailed information on considerations in performing a system PITR using the RESTORE SYSTEM utility.

In Figure1, the system restore process using an online copy process is demonstrated on a time line and shows the volume level copies being obtained using FlashCopy while the log is suspended. This ensures data consistency in the data base. In Figure 2 the copy is obtained after the SET LOG RESUME. This is because the EMC BCV process will ensure a 'crash consistent' copy and a suspended log is not necessary. During the restore process, all of the DB2 catalog and directory and user data are taken back by using this copy. The BSDSs and active logs stay as they were at the time the problem, requiring the PITR, was found. The system restore will start applying log records starting from the RBLP and going up to the SYSPITR / SYSPITRT point established in the BSDS.

FlashCopy - Log Suspend, Copy and Resume Flow

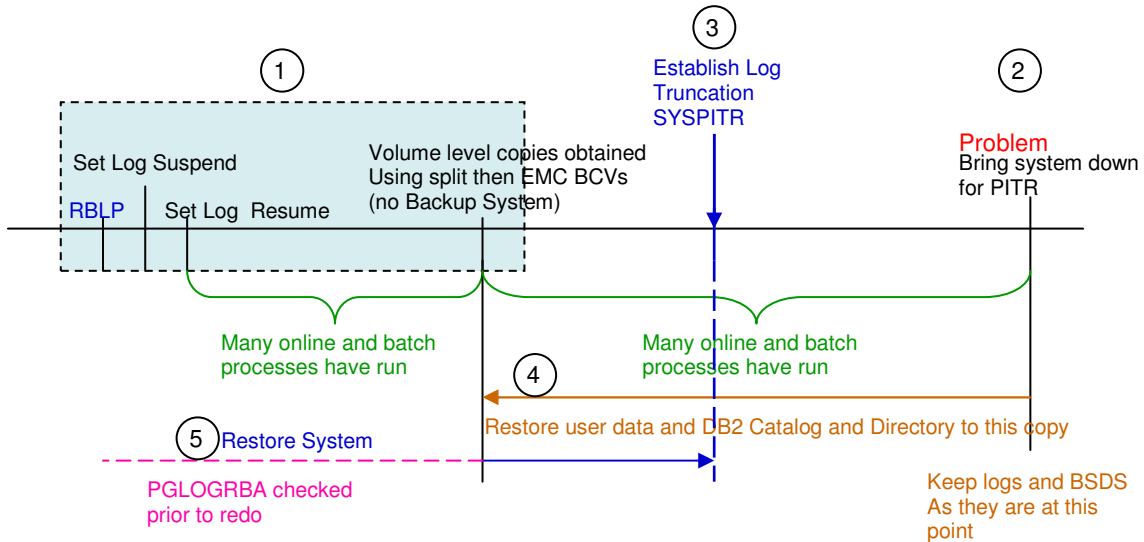


This Backup flow works and can be used by Restore System. .

- Suspend log
- Obtain volume level copies

Figure 1: Online FlashCopy process using log Suspend, obtaining the online copy using volume level copy technology, and log Resume.

EMC Copy after Set Log



This Backup flow will work:

- The SET LOG SUSPEND creates the RBLP
- With performing a split and then obtaining EMC BCVs the copy of the data is In a 'crash consistent' state.
- The logs that are being processed for reapply from the point of the RBLP up to the point the data was restored (the BCV copy) would not be reapplied because the PGLOGRBA in the data page is being checked against the log

Figure 2: EMC BCV process using log Suspend to create RBLP. Log is RESUME'd prior to obtaining copy because a suspended log it is not needed to obtain a 'crash consistent' copy using EMC BCV.

Using an offline copy in Point-in-time recovery (PITR)

When the DB2 system and its applications have been shutdown normally all units-of-recovery have been completed and written page have been externalized. The DB2 data is consistent. Taking a copy of the DB2 system and the application data when DB2 is shutdown normally is called an offline copy. There is no concern at this point in establishing data consistency, as we would in an online copy using Backup System utility or the SET LOG SUSPEND command.

However, the Restore System utility is still going to require a recovery base log point (RBLP) in order to perform a system level recovery. This can be accomplished by using the SET LOG SUSPEND command followed by a SET LOG RESUME command prior to the normal shutdown of the application and the DB2 system.

In the offline copy process you do not need to be concerned about what work might have run after the SET LOG RESUME and prior to the shutdown of the system, and what impact those log records would have if the offline copy is being used in a PITR. The log records that were created after the SET LOG RESUME and before the shutdown will not be reapplied to the data, because the PGLOGRBA of each page is checked before the log records are applied during the Restore System utility process.

In Figure 2 the backup flow will create consistent volume level copy, and will support the use of Restore System Utility for PITR:

- The FC is good because DB2 was down at the time it was obtained. So, one could take the whole system back to this copy if needed.
- Using this copy with Restore System would mean using the RBLP obtained prior to the shut down and consistent copy.
- The log that would be applied starting at the RBLP would have log records that have already been committed
- The PGLOGRBA of each page would be checked prior to applying the log records, and would not reapply log records created after the RBLP and prior to the shutdown.

Copy after Set Log Resume and DB2

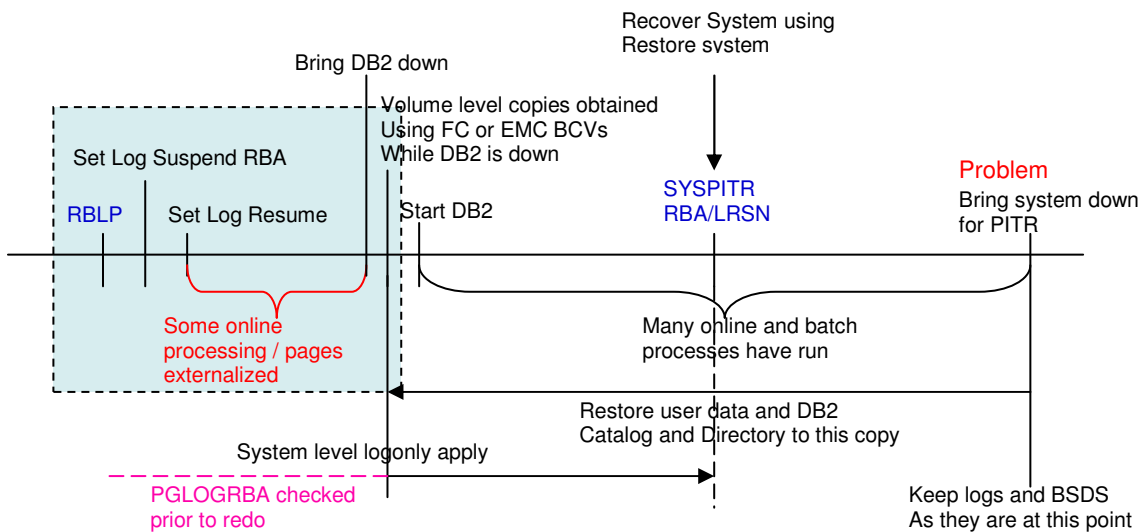


Figure 3: Offline Copy process using log Suspend/Resume to create RBLP, shutdown DB2, and obtaining the online copy using volume level copy technology.