Pivotal™

Rev: A02
Updated: December 16, 2013

Welcome to Pivotal Greenplum Database 4.3

Greenplum Database is a massively parallel processing (MPP) database server that supports next generation data warehousing and large-scale analytics processing. By automatically partitioning data and running parallel queries, it allows a cluster of servers to operate as a single database supercomputer performing tens or hundreds times faster than a traditional database. It supports SQL, MapReduce parallel processing, and data volumes ranging from hundreds of gigabytes, to hundreds of terabytes.

Note: This document contains pertinent release information about Greenplum Database 4.3. For previous versions of the release notes for Greenplum Database, go to Support Zone.

About Greenplum Database 4.3

Greenplum Database 4.3 is a major release that introduces a number of significant new features, as well as performance and stability enhancements. Please refer to the following sections for more information about this release.

- Product Enhancements
- Changed and Depricated Features
- Supported Platforms
- Resolved Issues in Greenplum Database 4.3
- Known Issues in Greenplum Database 4.3
- Upgrading to Greenplum Database 4.3
- Greenplum Database Tools Compatibility
- Greenplum Database Extensions Compatibility
- Hadoop Distribution Compatibility
- Greenplum Database 4.3 Documentation

Product Enhancements

Greenplum Database 4.3 includes enhancements in these areas:

- Greenplum Database High Availability
- Append-Optimized Tables
- Workfile Disk Spill Space Information

Greenplum Database High Availability

The Greenplum Database master mirroring feature has been enhanced. With master mirroring, a Greenplum Database backup master or *standby master* serves as a warm standby if the primary master becomes nonoperational.

- A Greenplum Database standby master is created while Greenplum Database is online. Greenplum Database does not need to be offline.
- Reboot of the standby master is not required when activating to the standby master to become primary master.
- Faster switching when Greenplum Database changes from the current active master to the standby master.

The following changes have been made to Greenplum Database:

- The Greenplum Database utilities `gpinitstandby` and `gpactivatestandby` have been changed.
- Greenplum Database administrative tables and views have been changed

For information about the changes to Greenplum Database utilities, and views and tables, see Changed and Depricated Features.

For information about the new server configuration parameters, see Server Configuration Parameters

For information about high availability and master mirroring, see the *Greenplum Database System Administration Guide*.

For information about Greenplum Database utilities, see the *Greenplum Database Utility Guide*.

Append-Optimized Tables

Append-optimized tables are similar to append-only tables and also allow `UPDATE` and `DELETE` operations on the table data. When migrating Greenplum Database from 4.2.x.x to 4.3, append-only tables are migrated to append-optimized tables.

Note: Migration of database with a large number of append-only tables might require a significant amount of time.

Append-optimized tables works best with denormalized fact tables in a data warehouse environment, where the data is static after it is loaded. Denormalized fact tables are typically the largest tables in the system. Fact tables are usually loaded in batches and accessed by read-only queries. Moving large fact tables to an append-only storage model eliminates the storage overhead of the per-row update visibility information, saving about 20 bytes per row.

IMPORTANT: Append-optimized tables cannot be used with the following functionality:

- Transactions with serializable isolation levels
- Updatable cursors

With append-optimized tables, you use the `VACUUM` command to reclaim the storage capacity from table data that was deleted or updated.

For information about the changes to the utilities, and views and tables, see Changed and Depricated Features.

For information about the new server configuration parameters, see Server Configuration Parameters

For information about creating append-optimized tables, see the `CREATE TABLE` command in the *Greenplum Database Reference Guide*.

For information about append-optimized tables and using `VACUUM` command to maintain append-optimized tables, see the *Greenplum Database Database Administrator Guide* and the *Greenplum Database System Administrator Guide.*

Workfile Disk Spill Space Information

In Greenplum Database 4.3 *gp_workfile_** views in the *gp_toolkit* administrative schema contain show information about all the queries that are currently using disk spill space.

Previously in 4.2.x.x releases, you created the views by running SQL scripts.

For information about the *gp_toolkit* administrative views, see the *Greenplum Database Reference Guide*.

Changed and Depricated Features

- Changed features
- Server Configuration Parameters
- Deprecated Features

Changed features

- Append-only tables are converted to append-optimized tables. Append-optimized tables are similar to append-only tables and also support `UPDATE` and `DELETE` operations. For information about append-optimized tables, see Append-Optimized Tables.

  The `CREATE TABLE` command has been updated to create append-optimized tables when the `WITH` clause contains the storage parameter `APPENDONLY=TRUE`.

  The `VACUUM` command has been updated to maintain append-optimized tables.

  The Greenplum Database system table *pg_appendonly* has been updated.

gp_toolkit schema has been updated with the following diagnostic functions that you can use to investigate the state of append-optimized tables

```
__gp_aoseg_name('table_name')
__gp_aoseg_history(oid)
__gp_aocsseg(oid)
__gp_aocsseg_history(oid)
__gp_aovisimap(oid)
__gp_aovisimap_hidden_info(oid)
__gp_aovisimap_entry(oid)
```

For information about the `CREATE TABLE` and `VACUUM` commands, system tables, and the *gp_toolkit* schema, see the *Greenplum Database Reference Guide*.

- Greenplum Database master mirroring has been enhanced to support easier creation of the Greenplum Database standby master and activation of the standby master to be come the primary master. For information about the master mirroring enhancements, see Greenplum Database High Availability.

  To support the enhanced functionality, changes have been made to the Greenplum Database utilities `gpinitstandby` and `gpactivatestandby`. Changes have also been made to Greenplum Database tables and views.

  - For the Greenplum Database utility `gpinitstandby` these options have been removed.

    `-L` (leave database stopped)

    `-M fast` (fast shutdown - rollback)

  - For the `gpinitstandby` utility, these options have been added

    `-P port` (specify a port for the standby master)

    `-F standby_filespaces` (specify file spaces for the standby master)

  - For the `gpactivatestandy` utility, this option has been removed.

    `-c new_standby_master_hostname`

  - These changes have been made to Greenplum Database administrative tables and views.

    The *gp_master_mirroring* table has been removed and has been replaced by the new view *pg_stat_replication.*

    The *pg_stat_activity* view has been modified.

  For information about the utilities see the *Greenplum Database Utility Guide.* For information about the tables and views, see the *Greenplum Database Reference Guide*.

- Inserting data into a partitioned table can only be done at parent partitioned table created with the `CREATE TABLE` command.

  When creating a partitioned table, Greenplum Database creates additional tables to manage the partitioning of data in a partitioned table. Using the `INSERT` command to insert data into tables created by Greenplum Database for use by a partitioned table is not allowed.

- The Greenplum Database *gp_workfile_** views have been added to the Greenplum Database administrative schema *gp_toolkit*. For information about the *gp_workfile_** views, see Workfile Disk Spill Space Information.

  For information about the *gp_toolkit* administrative views, see the *Greenplum Database Reference Guide.*

- For Greenplum Database 4.3, the file naming convention for Greenplum Database extension packages have changed. For information about the supported packages and the package naming convention, see Greenplum Database Extensions Compatibility.

Server Configuration Parameters

New Parameters

For information about server configuration parameters, see the *Greenplum Database Reference Guide*.

New Parameters

Master mirroring server configuration parameters

Table 1 New Server Configuration Parameters in 4.3 for Master Mirroring

Parameter Name	Value Range	Default Value	Description	Set Classifications
keep_wal_segments	0 - INT_MAX	5	For Greenplum Database master mirroring, sets the maximum number of processed WAL segment files that are saved by the by the active Greenplum Database master if a checkpoint operation occurs. The segment files are used to synchronize the active master on the standby master.	master system reload superuser
repl_catchup_within_range	0 - 64	1	For Greenplum Database master mirroring, controls updates to the active master. If the number of WAL segment files that have not been processed by the `walsender` exceeds this value, Greenplum Database updates the active master. If the number of segment files does not exceed the value, Greenplum Database blocks updates to the to allow the `walsender` process the files. If all WAL segments have been processed, the active master is updated.	master system reload superuser

Table 1 New Server Configuration Parameters in 4.3 for Master Mirroring

Parameter Name	Value Range	Default Value	Description	Set Classifications
replication_timeout	0 - INT_MAX	60000 ms (60 seconds)	For Greenplum Database master mirroring, sets the maximum time in milliseconds that the `walsender` process on the active master waits for a status message from the `walreceiver` process on the standby master. If a message is not received, the `walsender` logs an error message. The wal_receiver_status_interval controls the interval between `walreceiver` status messages.	master system reload superuser
wal_receiver_status_interval	integer 0-INT_MAX/1000	10 sec	For Greenplum Database master mirroring, sets the interval in seconds between `walreceiver` process status messages that are sent to the active master. Under heavy loads, the time might be longer. The value of replication_timeout controls the time that the `walsender` process waits for a `walreceiver` message.	master system reload superuser

Append-optimized server configuration parameters.

Table 2 New Server Configuration Parameters in 4.3 for Append-Optimized Tables

Parameter Name	Value Range	Default Value	Description	Set Classifications
gp_appendonly_compaction	Boolean	on	Enables compacting segment files during `VACUUM` commands. When disabled, `VACUUM` only truncates the segment files to the EOF value, as is the behavior of append-only tables in 4.2.x. The administrator might want to disable compaction in high I/O load situations or low space situations.	master session reload
gp_appendonly_compaction_threshold	integer (%)	10	Sets the threshold ratio (as a percentage) of hidden tuples to allow compaction of the segment file. If the ratio of hidden tuples in a segment file on a segment is less than this threshold, the segment file is not compacted on a `VACUUM FULL` call, and a LOG message is issued.	master session reload

Deprecated Features

These features have been deprecated for Greenplum Database master mirroring enhancements.

- These Greenplum Database utility `gpinitstandby` options have been removed.

  `-L (leave database stopped)`

  `-M fast (fast shutdown - rollback)`

- The Greenplum Database utility `gpactivatestandy` option `-c` *`new_standby_master_hostname`* has been removed
- The Greenplum Database system table *gp_master_mirroring* has been removed. The table is replaced by the new administrative view *pg_stat_replication*.

For information about Greenplum Database tables and views, see the *Greenplum Database Reference Guide*.

For information about Greenplum Database utilities, see the *Greenplum Database Utility Guide*.

For information about the master mirroring enhancements, see Greenplum Database High Availability.

Supported Platforms

Greenplum Database 4.3 runs on the following platforms:

- Red Hat Enterprise Linux 64-bit 5.5, 5.6, 5.7, 6.1, 6.2, and 6.4
- SUSE Linux Enterprise Server 64-bit 10 SP4, 11 SP1
- Solaris x86 64-bit v10 U7, U8, U9, U10
- Oracle Unbreakable Linux 64-bit 5.5
- CentOS 64-bit 5.5, 5.6, 5.7, 6.1, and 6.2

Greenplum Database 4.3 supports Data Domain Boost on Red Hat Enterprise Linux.

Greenplum Database 4.3 supports Data Domain Boost SDK version 2.4.2.2 with DDOS 5.0.1.0, 5.1 and 5.2.

GPDB 4.3 was certified against the following DCA versions:

- DCA v1 - ISO_VERSION="1.2.1.1"
- DCA v2 - ISO_VERSION="2.0.2.0"

Note: For the most recent server platform support information for Greenplum Database, check the EMC Support Matrix (ESM).

To access the ESM, go to the Support Zone home page and click the link **E-Lab Interoperability Navigator**.

In the E-Lab Interoperability Navigator, search for Greenplum Database and add one or more search results to the search cart. Click **Get Results** to display links to EMC Support Statements.

Resolved Issues in Greenplum Database 4.3

The table below lists issues that are now resolved in Greenplum Database 4.3.

For issues resolved in prior releases, refer to the corresponding release notes available from Support Zone.

Table 3 Resolved Issues in 4.3

Issue Number	Category	Description
21522	Backup and Restore	The Greenplum Database utility pg_dump printed information-level messages (messages with the label [INFO]) to stderr that were not printed in previous releases. These messages were printed even when pg_dump completes without errors.

Known Issues in Greenplum Database 4.3

This section lists the known issues in Greenplum Database 4.3. A workaround is provided where applicable.

For known issues discovered in previous releases, including patch releases to Greenplum Database 4.2.x, 4.1 or 4.0.x, see the corresponding release notes, available from Support Zone:

Table 4 All Known Issues in 4.3

Issue	Category	Description
21917	Replication: Segment Mirroring	**In some rare cases after the Greenplum Database utility gprecoverseg was run, some append-optimized tables and a persistent table were detected having less data on a mirror segment corresponding to a primary segment.**

20453	Query Planner	**For SQL queries of either of the following forms:**
		`SELECT columns FROM table WHERE table.column NOT IN subquery;`
		`SELECT columns FROM table WHERE table.column = ALL subquery;`
		tuples that satisfy both of the following conditions are not included in the result set:
		• *table.column* **is NULL.**
		• *subquery* **returns the empty result.**

21724	Query Planner	**Greenplum Database executes an SQL query in two stages if a scalar subquery is involved. The output of the first stage plan is fed into the second stage plan as a external parameter. If the first stage plan generates zero tuples and directly contributes to the output of the second stage plan, incorrect results might be returned.**
21838	Backup and Restore	**When restoring sets of tables with the Greenplum Database utility gpdbrestore, the table schemas must be defined in the database. If a table's schema is not defined in the database, the table is not restored. When performing a full restore, the database schemas are created when the tables are restored.** **Workaround:** Before restoring a set of tables, create the schemas for the tables in the database.
21129	DDL and Utility Statements	**SSL is only supported on the master host. It is not supported on segment hosts.**
20822	Backup and Restore	**Special characters such as !, $, #, and @ cannot be used in the password for the Data Domain Boost user when specifying the Data Domain Boost credentials with the `gpcrondump` options `--ddboost-host` and `--ddboost-user`.**
18247	DDL and Utility Statements	**`TRUNCATE` command does not remove rows from a sub-table of a partitioned table. If you specify a sub-table of a partitioned table with the `TRUNCATE` command, the command does not remove rows from the sub-table and its child tables.** **Workaround**: Use the `ALTER TABLE` command with the `TRUNCATE PARTITION` clause to remove rows from the sub-table and its child tables.
19788	Replication: Resync, Transaction Management	**In some rare circumstances, performing a full recovery with `gprecoverseg` fails due to inconsistent LSN.** **Workaround**: Stop and restart the database. Then perform a full recovery with `gprecoverseg`.

Table 4 All Known Issues in 4.3

Issue	Category	Description
19772	Interconnect	**After installing Greenplum Database 4.2.4, downgrading the Greenplum Database installation to a previous minor version causes a crash.** **Workaround**: Before downgrading Greenplum Database 4.2.4 to a previous minor version, change the value for the Greenplum Database parameter `gp_interconnect_type` from `UDPIFC` to a supported value such as `UDP`. The parameter value `UDPIFC` was introduced in Greenplum Database 4.2.4 and is not valid in previous versions.
19705	Loaders: gpload	**`gpload` fails on Windows XP with Python 2.6.** **Workaround**: Install Python 2.5 on the system where gpload is installed.
19493 19464 19426	Backup and Restore	**The `gpcrondump` and `gpdbrestore` utilities do not handle errors returned by DD Boost or Data Domain correctly.** These are two examples: • If invalid Data Domain credentials are specified when setting the Data Domain Boost credentials with the `gpcrondump` utility, the error message does not indicate that invalid credentials were specified. • Restoring a Greenplum database from a Data Domain system with `gpdbrestore` and the `--ddboost` option indicates success even though segment failures occured during the restore. **Workaround**: The errors are logged in the master and segment server backup or restore status and report files. Scan the status and report files to check for error messages.
19278	Backup and Restore	**When performing a selective restore of a partitioned table from a full backup with `gpdbrestore`, the data from leaf partitions are not restored.** **Workaround**: When doing a selective restore from a full backup, specify the individual leaf partitions of the partitioned tables that are being restored. Alternatively, perform a full backup, not a selective backup.
16129	Management Scripts Suite	**`gpkill` does not run on the Solaris platform.** The gpkill utility is using an internal tool called “glider” to introspect processes and glean/archive some relevant information before actually killing processes. In some cases, our invocation of this tool fails to yield the desired introspective information.
15692 17192	Backup and Restore	**Greenplum Database’s implementation of RSA lock box for Data Domain Boost changes backup and restore requirements for customers running SUSE.** The current implementation of the RSA lock box for Data Domain Boost login credential encryption only supports customers running on Red Hat Enterprise Linux. **Workaround**: If you run Greenplum Database on SUSE, use NFS as your backup solution. See the *Greenplum Database System Administrator Guide* for information on setting up a NFS backup.
18850	Backup and Restore	Data Domain Boost credentials cannot be set up in some environments due to the absence of certain libraries (for example, `libstdc++`) expected to reside on the platform. **Workaround**: Install the missing libraries manually on the system.
18851	Backup and Restore	When performing a data-only restore of a particular table, it is possible to introduce data into Greenplum Database that contradicts the distribution policy of that table. In such cases, subsequent queries may return unexpected and incorrect results. To avoid this scenario, we suggest you carefully consider the table schema when performing a restore.
18774	Loaders	External web tables that use IPv6 addresses must include a port number.

Table 4 All Known Issues in 4.3

Issue	Category	Description
18713	Catalog and Metadata	Drop language plpgsql cascade results in a loss of `gp_toolkit` functionality. **Workaround**: Reinstall `gp_toolkit`.
18710	Management Scripts Suite	Greenplum Management utilities cannot parse IPv6 IP addresses. **Workaround**: Always specify IPv6 hostnames rather than IP addresses
18703	Loaders	The bytenum field (byte offset in the load file where the error occurred) in the error log when using gpfdist with data in text format errors is not populated, making it difficult to find the location of an error in the source file.
12468	Management Scripts Suite	`gpexpand --rollback` fails if an error occurs during expansion such that it leaves the database down `gpstart` also fails as it detects that expansion is in progress and suggests to run `gpexpand --rollback` which will not work because the database is down. **Workaround**: Run `gpstart -m` to start the master and then run rollback,
18785	Loaders	Running `gpload` with the `--ssl` option and the relative path of the source file results in an error that states the source file is missing. **Workaround**: Provide the full path in the yaml file or add the loaded data file to the certificate folder.
18414	Loaders	Unable to define external tables with fixed width format and empty line delimiter when file size is larger than gpfdist chunk (by default, 32K).
14640	Backup and Restore	`gpdbrestore` outputting incorrect non-zero error message. When performing single table restore, `gpdbrestore` gives warning messages about non-zero tables however prints out zero rows.
17285	Backup and Restore	NFS backup with `gpcrondump -c` can fail. In circumstances where you haven't backed up to a local disk before, backups to NFS using `gpcrondump` with the `-c` option can fail. On fresh systems where a backup has not been previously invoked there are no dump files to cleanup and the `-c` flag will have no effect. **Workaround**: Do not run `gpcrondump` with the `-c` option the first time a backup is invoked from a system.
17837	Upgrade/ Downgrade	Major version upgrades internally depend on the `gp_toolkit` system schema. The alteration or absence of this schema may cause upgrades to error out during preliminary checks. **Workaround**: To enable the upgrade process to proceed, you need to reinstall the `gp_toolkit` schema in all affected databases by applying the SQL file found here: `$GPHOME/share/postgresql/gp_toolkit.sql`.
17513	Management Scripts Suite	Running more than one `gpfilespace` command concurrently with itself to move either temporary files (`--movetempfilespace`) or transaction files (`--movetransfilespace`) to a new filespace can in some circumstances cause OID inconsistencies. **Workaround**: Do not run more than one `gpfilespace` command concurrently with itself. If an OID inconsistency is introduced `gpfilespace --movetempfilespace` or `gpfilespace --movetransfilespace` can be used to revert to the default filespace.

Table 4 All Known Issues in 4.3

Issue	Category	Description
17780	DDL/DML: Partitioning	`ALTER TABLE ADD PARTITION` inheritance issue When performing an `ALTER TABLE ADD PARTITION` operation, the resulting parts may not correctly inherit the storage properties of the parent table in cases such as adding a default partition or more complex subpartitioning. This issue can be avoided by explicitly dictating the storage properties during the `ADD PARTITION` invocation. For leaf partitions that are already afflicted, the issue can be rectified through use of `EXCHANGE PARTITION`.
17795	Management Scripts Suite	Under some circumstances, `gppkg` on SUSE is unable to correctly interpret error messages returned by rpm. On SUSE, `gppkg` is unable to operate correctly under circumstances that require a non-trivial interpretation of underlying rpm commands. This includes scenarios that result from overlapping packages, partial installs, and partial uninstalls.
17604	Security	A Red Hat Enterprise Linux (RHEL) 6.x security configuration file limits the number of processes that can run on gpadmin. RHEL 6.x contains a security file (/etc/security/limits.d/90-nproc.conf) that limits available processes running on gpadmin to 1064. **Workaround**: Remove this file or increase the processes to 131072.
17415	Installer	When you run `gppkg -q -info<some gppkg>`, the system shows the GPDBversion as main build dev.
17334	Management Scripts Suite	You may see warning messages that interfere with the operation of management scripts when logging in. Greenplum recommends that you edit the /etc/motd file and add the warning message to it. This will send the messages to are redirected to `stdout` and not `stderr`. You must encode these warning messages in UTF-8 format.
17221	Resource Management	Resource queue deadlocks may be encountered if a cursor is associated with a query invoking a function within another function.
17113	Management Scripts Suite	Filespaces are inconsistent when the Greenplum database is down. Filespaces become inconsistent in case of a network failure. Greenplum recommends that processes such as moving a filespace be done in an environment with an uninterrupted power supply.
17189	Loaders: gpfdist	`gpfdist` shows the error “Address already in use” after successfully binding to socket IPv6. Greenplum supports IPv4 and IPv6. However, `gpfdist` fails to bind to socket IPv4, and shows the message “Address already in use”, but binds successfully to socket IPv6.
16278	Management Scripts Suite	`gpkill` shows that it failed to kill the gpload process, but in fact the process was successfully aborted with all the data loaded correctly.
16269	Management Scripts Suite	`gpkill` should attempt to kill each given pid. `gpkill` accepts the list of pids but only shows that one of the processes may not be killed.

Table 4 All Known Issues in 4.3

Issue	Category	Description

16519	Backup and Restore	Limited data restore functionality and/or restore performance issues can occur when restoring tables from a full database backup where the default backup directory was not used.
		In order to restore from backup files not located in the default directory you can use the `-R` to point to another host and directory. This is not possible however, if you want to point to a different directory on the same host (NFS for example).
		Workaround: Define a symbolic link from the default dump directory to the directory used for backup, as shown in the following example:
		1. Perform a full Database Backup to a specific NFS directory:
		`$ gpcrondump -x <db_name> -z -u /backup/DCA-93 -a`
		2. Create a file listing the segment servers:
		`$ vi /home/gpadmin/segments`
		`sdw1`
		`sdw2`
		`sdw3`
		`….`
		3. Remove the relevant date folder from db_dumps directories on the master and segments:
		`$ rm -r /data/master/gpseg-1/db_dumps/20120619`
		`$ gpssh -f segments 'rm -r /data1/primary/gpseg*/db_dumps/20120619'`
		`$ gpssh -f segments 'rm -r /data2/primary/gpseg*/db_dumps/20120619'`
		4. Create a symbolic link between the master and segment directories and the directory to which you backed up in step 1. Only the master and sdw1 was shown here, write a script for the remaining segments:
		`$ ln -s /backup/DCA-93/db_dumps/20120619 /data/master/gpseg-1/db_dumps/20120619`
		`$ gpssh -h sdw1 'ln -s /backup/DCA-93/db_dumps/20120619 /data1/primary/gpseg0/db_dumps/20120619'`
		`$ gpssh -h sdw1 'ln -s /backup/DCA-93/db_dumps/20120619 /data1/primary/gpseg1/db_dumps/20120619'`
		`$ gpssh -h sdw1 'ln -s /backup/DCA-93/db_dumps/20120619 /data1/primary/gpseg2/db_dumps/20120619'`
		`$ gpssh -h sdw1 'ln -s /backup/DCA-93/db_dumps/20120619 /data2/primary/gpseg3/db_dumps/20120619'`
		`$ gpssh -h sdw1 'ln -s /backup/DCA-93/db_dumps/20120619 /data2/primary/gpseg4/db_dumps/20120619'`
		`$ gpssh -h sdw1 'ln -s /backup/DCA-93/db_dumps/20120619 /data2/primary/gpseg5/db_dumps/20120619'`
		5. Restore from backup files:
		`$ gpdbrestore -t 20120619061835 -T <schema.table> -a`
		6. Remove the symbolic links
		`$ rrm -r /data/master/gpseg-1/db_dumps/20120619`
		`$ gpssh -f segments 'rm -r /data1/primary/gpseg*/db_dumps/20120619'`
		`$ gpssh -f segments 'rm -r /data2/primary/gpseg*/db_dumps/20120619'`

Table 4 All Known Issues in 4.3

Issue	Category	Description
16267 15954	Management Scripts Suite	`gpkill` cannot kill processes that are deemed STUCK. **Workaround**: Kill the STUCK processes using OS kill.
16067	Management Scripts Suite	`gpkill` does not validate the user input for `password_hash_algorithm` The current behavior shows a success message for any input value. However, the server configuration parameter value is not updated if the input is invalid. When the user tries to set the value for a session from within psql, it fails with the appropriate error message.
16064	Backup and Restore	Restoring a compressed dump with the `--ddboost` option displays incorrect dump parameter information. When using `gpdbrestore --ddboost` to restore a compressed dump, the restore parameters incorrectly show “`Restore compressed dump = Off`”. This error occurs even if `gpdbrestore` passes the `--gp-c` option to use gunzip for in-line de-compression.
15899	Backup and Restore	When running `gpdbrestore` with the list (`-L)` option, external tables do not appear; this has no functional impact on the restore job.

Upgrading to Greenplum Database 4.3

The upgrade path supported for this release is Greenplum Database 4.2.x.x to Greenplum Database 4.3. The minimum recommended upgrade path for this release is from Greenplum Database version 4.2.x.x. If you have an earlier major version of the database, you must first upgrade to version 4.2.x.x.

For detailed upgrade procedures and information, see the following sections:

- Upgrading from 4.2.x.x to 4.3
- For Users Running Greenplum Database 4.1.x.x
- For Users Running Greenplum Database 4.0.x.x
- For Users Running Greenplum Database 3.3.x.x
- Troubleshooting a Failed Upgrade

If you are utilizing Data Domain Boost, you have to re-enter your DD Boost credentials after upgrading from Greenplum Database 4.2.x.x to 4.3 as follows:

```
gpcrondump --ddboost-host ddboost_hostname --ddboost-user
ddboost_user
```

Note that if you do not reenter your login credentials after an upgrade, your backup will never start because the Greenplum Database cannot connect to the Data Domain system. You will receive an error advising you to check your login credentials.

Upgrading from 4.2.x.x to 4.3

This section describes how you can upgrade from Greenplum Database 4.2.x.x or later to Greenplum Database 4.3. For users running versions prior to 4.2.x.x of Greenplum Database, see the following:

- For Users Running Greenplum Database 4.1.x.x
- For Users Running Greenplum Database 4.0.x.x
- For Users Running Greenplum Database 3.3.x.x

Planning Your Upgrade

Before you begin your upgrade, make sure the master and all segments (data directories and filespace) have at least 2GB of free space.

Prior to upgrading your database, Pivotal recommends that you run a pre-upgrade check to verify your database is healthy.

You can perform a pre-upgrade check by executing the `gpmigrator` (`_mirror`) utility with the `--check-only` option.

For example:

```
source $new_gphome/greenplum_path.sh;
gpmigrator_mirror --check-only $old_gphome $new_gphome
```

Note: Performing a pre-upgrade check of your database with the `gpmigrator` (`_mirror`) utility should done during a database maintenance period. When the utility checks the database catalog, users cannot access the database.

Migrating a Greenplum Database That Contains AO Tables

The migration process updates AO tables that are in a Greenplum Database to UAO tables. For a database that contains a large number of AO tables, the conversion to UAO tables might take a considerable amount of time.

Upgrade Procedure

This section divides the upgrade into the following phases: pre-upgrade preparation, software installation, upgrade execution, and post-upgrade tasks.

We have also provided you with an Upgrade Checklist that summarizes this procedure.

IMPORTANT: Carefully evaluate each section and perform all required and conditional steps. Failing to perform any of these steps can result in an aborted upgrade, placing your system in an unusable or even unrecoverable state.

Pre-Upgrade Preparation (on your 4.2.x system)

Perform these steps on your current 4.2.x Greenplum Database system. This procedure is performed from your Greenplum master host and should be executed by the Greenplum superuser (`gpadmin`).

1. Log in to the Greenplum Database master as the `gpadmin` user:

   ```
   $ su - gpadmin
   ```

2. (*optional*) Vacuum all databases prior to upgrade. For example:

   ```
   $ vacuumdb database_name
   ```

3. (*optional*) Clean out old server log files from your master and segment data directories. For example, to remove log files from 2011 from your segment hosts:

   ```
   $ gpssh -f seg_host_file -e 'rm
   /gpdata/*/gp*/pg_log/gpdb-2011-*.csv'
   ```

   Note: Running Vacuum and cleaning out old logs files is not required, but it will reduce the size of Greenplum Database files to be backed up and migrated.

4. Run `gpstate` to check for failed segments.

   ```
   $ gpstate
   ```

5. If you have failed segments, you must recover them using `gprecoverseg` before you can upgrade.

   ```
   $ gprecoverseg
   ```

   Note: It might be necessary to restart the database if the preferred role does not match the current role; for example, if a primary segment is acting as a mirror segment or a mirror segment is acting as a primary segment.

6. Copy or preserve any additional folders or files (such as backup folders) that you have added in the Greenplum data directories or `$GPHOME` directory. Only files or folders strictly related to Greenplum Database operations are preserved by the migration utility.

Install the Greenplum Database 4.3 Software Binaries

1. Download or copy the installer file to the Greenplum Database master host.

2. Unzip the installer file. For example:

```
# unzip greenplum-db-4.3-PLATFORM.zip
```

3. Launch the installer using `bash`. For example:

```
# /bin/bash greenplum-db-4.3-PLATFORM.bin
```

4. The installer will prompt you to accept the Greenplum Database license agreement. Type `yes` to accept the license agreement.

5. The installer will prompt you to provide an installation path. Press `ENTER` to accept the default install path (for example: `/usr/local/greenplum-db-4.3`), or enter an absolute path to an install location. You must have write permissions to the location you specify.

6. The installer installs the Greenplum Database software and creates a `greenplum-db` symbolic link one directory level above your version-specific Greenplum installation directory. The symbolic link is used to facilitate patch maintenance and upgrades between versions. The installed location is referred to as `$GPHOME`.

7. Source the path file from your new 4.3 installation. For example:

```
$ source /usr/local/greenplum-db-4.3/greenplum_path.sh
```

8. Run the `gpseginstall` utility to install the 4.3 binaries on all the segment hosts specified in the *`hostfile`*. For example:

```
$ gpseginstall -f hostfile
```

Upgrade Execution

During upgrade, all client connections to the master will be locked out. Inform all database users of the upgrade and lockout time frame. From this point onward, users should not be allowed on the system until the upgrade is complete.

9. Source the path file from your old 4.2.x.x installation. For example:

```
$ source /usr/local/greenplum-db-4.2.6.3/greenplum_path.sh
```

10. (*optional but strongly recommended*) Back up all databases in your Greenplum Database system using `gpcrondump` (or zfs snapshots on Solaris systems). See the *Greenplum Database Administrator Guide* for more information on how to do backups using `gpcrondump`. Make sure to secure your backup files in a location outside of your Greenplum data directories.

11. If your system has a standby master host configured, remove the standby master from your system configuration. For example:

```
$ gpinitstandby -r
```

12. Perform a clean shutdown of your current Greenplum Database 4.2.x.x system. For example:

```
$ gpstop
```

13. Source the path file from your new 4.3 installation. For example:

```
$ source /usr/home/greenplum-db-4.3/greenplum_path.sh
```

14. Update the Greenplum Database environment so it is referencing your new 4.3 installation.

   a. For example, update the `greenplum-db` symbolic link on the master and standby master to point to the new 4.3 installation directory. For example (as root):

   ```
   # rm -rf /usr/local/greenplum-db
   # ln -s /usr/local/greenplum-db-4.3 /usr/local/greenplum-db
   # chown -R gpadmin /usr/local/greenplum-db
   ```

   b. Using `gpssh`, also update the `greenplum-db` symbolic link on all of your segment hosts. For example (as root):

   ```
   # gpssh -f segment_hosts_file
   => rm -rf /usr/local/greenplum-db
   => ln -s /usr/local/greenplum-db-4.3
   /usr/local/greenplum-db
   => chown -R gpadmin /usr/local/greenplum-db
   => exit
   ```

15. (*optional but recommended*) Prior to running the migration, perform a pre-upgrade check to verify that your database is healthy by executing the 4.3 version of the `gpmigrator` utility with the `--check-only` option. For example:

```
# gpmigrator_mirror --check-only
   /usr/local/greenplum-db-4.2.6.3
   /usr/local/greenplum-db-4.3
```

16. As `gpadmin`, run the 4.3 version of the migration utility specifying your old and new `GPHOME` locations. If your system has mirrors, use `gpmigrator_mirror`. If your system does not have mirrors, use `gpmigrator`. For example on a system with mirrors:

```
$ su - gpadmin
$ gpmigrator_mirror /usr/local/greenplum-db-4.2.6.3
   /usr/local/greenplum-db-4.3
```

Note: If the migration does not complete successfully, contact Customer Support (see "Troubleshooting a Failed Upgrade" on page 23).

17. The migration can take a while to complete. After the migration utility has completed successfully, the Greenplum Database 4.3 system will be running and accepting connections.

Note: After the migration utility has completed, the resynchronization of the mirror segments with the primary segments continues. Even though the system is running, the mirrors are not active until the resynchronization is complete.

Post-Upgrade (on your 4.3 system)

18. If your system had a standby master host configured, reinitialize your standby master using `gpinitstandby`:

```
$ gpinitstandby -s standby_hostname
```

19. If your system uses external tables with `gpfdist`, stop all `gpfdist` processes on your ETL servers and reinstall `gpfdist` using the compatible Greenplum Database 4.3 Load Tools package. Application Packages are available at the EMC Download Center.

20. Rebuild any custom modules against your 4.3 installation (for example, any shared library files for user-defined functions in `$GPHOME/lib`).

21. Use the Greenplum Database `gppkg` utility to install Greenplum Database extensions. If you were previously using any Greenplum Database extensions such as pgcrypto, PL/R, PL/Java, PL/Perl, and PostGIS, download the corresponding packages from the EMC Download Center, and install using this new utility. See the *Greenplum Database Administrator Guide 4.3* for usage details.

22. If you want to utilize the Greenplum Command Center management tool, install the latest Command Center Console and update your environment variable to point to the latest Command Center binaries (source the `gpperfmon_path.sh` file from your new installation).

Note that the Greenplum Command Center management tool replaces Greenplum Performance Monitor.

Command Center Console packages are available from the EMC Download Center.

23. Inform all database users of the completed upgrade. Tell users to update their environment to source the Greenplum Database 4.3 installation (if necessary).

Upgrade Checklist

This checklist provides a quick overview of all the steps required for an upgrade from 4.2.x.x to 4.3. Detailed upgrade instructions are provided in the Upgrade Procedure section.

Pre-Upgrade Preparation (on your current system)	
* 4.2.x.x system is up and available	
□	Log in to your master host as the `gpadmin` user (your Greenplum superuser).
□	(Optional) Run `VACUUM` on all databases,
□	(Optional) Remove old server log files from `pg_log` in your master and segment data directories.
□	Check for and recover any failed segments (`gpstate`, `gprecoverseg`).
□	Copy or preserve any additional folders or files (such as backup folders).
□	Install the Greenplum Database 4.3 binaries on all Greenplum hosts.
□	Inform all database users of the upgrade and lockout time frame.
Upgrade Execution	
* The system will be locked down to all user activity during the upgrade process	
□	Backup your current databases.
□	Remove the standby master (`gpinitstandby -r`).
□	Do a clean shutdown of your current system (`gpstop`).
□	Update your environment to source the new Greenplum Database 4.3 installation.
□	Run the upgrade utility (`gpmigrator_mirror` if you have mirrors, `gpmigrator` if you do not).
□	After the upgrade process finishes successfully, your 4.3 system will be up and running.
Post-Upgrade (on your 4.3 system)	
* The 4.2.x.x system is up	
□	Reinitialize your standby master host (`gpinitstandby`).

□	Upgrade `gpfdist` on all of your ETL hosts.
□	Rebuild any custom modules against your 4.3 installation.
□	Download and install any Greenplum Database extensions.
□	(Optional) Install the latest Command Center Console and update your environment to point to the latest Command Center binaries.
□	Inform all database users of the completed upgrade.

For Users Running Greenplum Database 4.1.x.x

Users on a release prior to 4.1.x.x cannot upgrade directly to 4.3.

1. Upgrade from your current release to 4.2.x.x (follow the upgrade instructions in the latest Greenplum Database 4.2.x.x release notes available on Support Zone).

2. Follow the upgrade instructions in these release notes for Upgrading from 4.2.x.x to 4.3.

For Users Running Greenplum Database 4.0.x.x

Users on a release prior to 4.1.x.x cannot upgrade directly to 4.3.

1. Upgrade from your current release to 4.1.x.x (follow the upgrade instructions in the latest Greenplum Database 4.1.x.x release notes available on Support Zone).

2. Upgrade from the current release to 4.2.x.x (follow the upgrade instructions in the latest Greenplum Database 4.2.x.x release notes available on Support Zone).

3. Follow the upgrade instructions in these release notes for Upgrading from 4.2.x.x to 4.3.

For Users Running Greenplum Database 3.3.x.x

Users on a release prior to 4.0.x.x cannot upgrade directly to 4.3.

1. Upgrade from your current release to the latest 4.0.x.x release (follow the upgrade instructions in the latest Greenplum Database 4.0.x.x release notes available on Support Zone).

2. Upgrade the 4.0.x.x release to the latest 4.1.x.x release (follow the upgrade instructions in the latest Greenplum Database 4.1.x.x release notes available on Support Zone).

3. Upgrade from the 4.1.1 release to the latest 4.2.x.x release (follow the upgrade instructions in the latest Greenplum Database 4.2.x.x release notes available on Support Zone).

4. Follow the upgrade instructions in these release notes for Upgrading from 4.2.x.x to 4.3.

Troubleshooting a Failed Upgrade

If you experience issues during the migration process, go to the Support page at Support Zone or contact Greenplum customer support at one of the following numbers:

United States: 800-782-4362 (1-800-SVC-4EMC)

Canada: 800-543-4782

Worldwide: +1-508-497-7901

Be prepared to provide the following information:

- A completed Upgrade Procedure.
- Log output from `gpmigrator` and `gpcheckcat` (located in `~/gpAdminLogs`)

Greenplum Database Tools Compatibility

Client Tools

Greenplum releases a number of client tool packages on various platforms that can be used to connect to Greenplum Database and the Greenplum Command Center management tool. The following table describes the compatibility of these packages with this Greenplum Database release.

Note: Release and support for AIX Clients/Connectivity/Loaders is pending. We will update the Release Notes and the EMC Download Center when available.

Tool packages are available from the EMC Download Center.

Table 5 Greenplum Database Tools Compatibility

Client Package	Description of Contents	Client Version	Server Versions
Greenplum Clients	Greenplum Database Command-Line Interface (psql) Greenplum MapReduce (gpmapreduce)[1]	4.3	4.3
Greenplum Connectivity	Standard PostgreSQL Database Drivers (ODBC, JDBC) PostgreSQL Client C API (libpq)	4.3	4.3
Greenplum Loaders	Greenplum Database Parallel Data Loading Tools (gpfdist, gpload)	4.3	4.3
Greenplum Command Center	Greenplum Database management tool.	1.2.0.1	4.3

[1] gpmapreduce is not available on Windows.

Greenplum GPText

GPText enables processing mass quantities of raw text data (such as social media feeds or e-mail databases) into mission-critical information that guides business and project decisions. GPText joins the Greenplum Database massively parallel-processing database server with Apache Solr enterprise search.

GPText requires Greenplum Database. See the GPText release notes for the required version of Greenplum Database.

Greenplum Database Extensions Compatibility

Greenplum Database delivers an agile, extensible platform for in-database analytics, leveraging the system's massively parallel architecture. Greenplum Database enables turn-key in-database analytics with Greenplum extensions.

You can download Greenplum extensions packages from the EMC Download Center and install them using the Greenplum Packager Manager (`gppkg`). See the *Greenplum Database Administrator Guide 4.3* for details.

Note that Greenplum Package Manager installation files for extension packages may release outside of standard Database release cycles. Therefore, for the latest install and configuration information regarding any supported database package/extension, go to the `Support` site and download `Primus Article 288189` from our knowledge base (Requires a valid login to the EMC Support site).

The following table provides information about the compatibility of the Greenplum Database Extensions and their components with this Greenplum Database release.

Note that the PL/Python database extension is already included with the standard Greenplum database distribution.

Table 6 Greenplum Database Extensions Compatibility

Greenplum Database Extension	Extension Components	
	Name	**Version**
PostGIS 2.0 for Greenplum Database 4.3.x.x	PostGIS	2.0.3
	Proj	4.8.0
	Geos	3.3.8
PostGIS 1.0 for Greenplum Database	PostGIS	1.4.2
	Proj	4.7.0
	Geos	3.2.2
PL/Java 1.0 for Greenplum Database 4.3.x.x	PL/Java	Based on 1.4.0
	Java JDK	1.6.0_26 Update 31
PL/R 1.0 for Greenplum Database 4.3.x.x	PL/R	8.3.0.12
	R	2.13.0
PL/Perl 1.2 for Greenplum Database 4.3.x.x	PL/Perl	Based on PostgreSQL 9.1
	Perl	5.12.4 on RHEL 6.x 5.5.8 on RHEL 5.x, SUSE 10

Table 6 Greenplum Database Extensions Compatibility

Greenplum Database Extension	Extension Components	
	Name	**Version**
PL/Perl 1.1 for Greenplum Database	PL/Perl	Based on PostgreSQL 9.1
	Perl	5.12.4 on RHEL 5.x, SUSE 10
PL/Perl 1.0 for Greenplum Database	PL/Perl	Based on PostgreSQL 9.1
	Perl	5.12.4 on RHEL 5.x, SUSE 10
Pgcrypto 1.1 for Greenplum Database 4.3.x.x	Pgcrypto	Based on PostgreSQL 8.3
Greenplum Hadoop File System	gphdfs	1.1
	gphdfs	1.2
	gphdfs	1.3
	gphdfs	1.4
MADlib for Greenplum Database 4.3.x.x	MADlib	Based on MADlib version 1.4

Greenplum Database 4.3 supports these minimum Greenplum Database extensions package versions.

Table 7 Greenplum Database 4.3 Package Version

Greenplum Database Extension	Minimum Package Version
PostGIS	2.0
PL/Java	1.1
PL/Perl	1.2
PL/R	1.0
Pgcrypto	1.1
gphdfs	1.5.1
MADlib	1.8

Package File Naming Convention

For Greenplum Database 4.3, this is the package file naming format.

*pkgname-ver*_**pv***pkg-version*_**gpdb***rel-OS-version-arch*.**gppkg**

This example is the package name for a postGIS package.

```
postgis-ossv2.0.3_pv2.0_gpdb4.3-rhel5-x86_64.gppkg
```

pkgname-ver - The package name and optional version of the software that was used to create the package extension. If the package is based on open source software, the version has format **ossv***version*. The *version* is the version of the open source software that the package is based on. For the postGIS package, `ossv2.0.3` specifies that the package is based on postGIS version 2.0.3.

pv*pkg-version* - The package version. The version of the Greenplum Database package. For the postGIS package, `pv2.0` specifies that the Greenplum Database package version is 2.0.

gpdb*rel-OS-version-arch* - The compatible Greenplum Database release. For the postGIS package, `gpdb4.3-rhel5-x86_64` specifies that package is compatible with Greenplum Database 4.3 on Red Hat Enterprise Linux version 5.x, x86 64-bit architecture.

Hadoop Distribution Compatibility

Use the `gppkg` utility to install the gNet package containing the jar file for the extensions, the libraries, and the documentation for the gphdfs extensions. To install the correct distribution, refer to the following Hadoop extensions compatibility matrix:

Table 8 Hadoop Extensions Compatibility

Hadoop Distribution	Version
Pivotal HD	Pivotal HD 1.0[1]
Greenplum HD	Greenplum HD 1.1
	Greenplum HD 1.2
Cloudera	cdh3u2
	cdh3u4
	CDH4.1 with MRv1
Greenplum MR	Greenplum MR 1.0
	Greenplum MR 1.2

[1] A distribution of Hadoop 2.0

Greenplum Database 4.3 Documentation

For the latest Greenplum Database documentation go to Support Zone. Greenplum documentation is provided in PDF format.

Table 9 Greenplum Database Documentation

Title	Revision
Greenplum Database 4.3 Release Notes	A02
Greenplum Database 4.3 Installation Guide	A01
Greenplum Database 4.3 Database Administrator Guide	A01
Greenplum Database 4.3 System Administrator Guide	A01
Greenplum Database 4.3 Reference Guide	A01
Greenplum Database 4.3 Utility Guide	A01
Greenplum Database 4.3 Client Tools for UNIX	A01
Greenplum Database 4.3 Client Tools for Windows	A01
Greenplum Database 4.3 Connectivity Tools for UNIX	A01
Greenplum Database 4.3 Connectivity Tools for Windows	A01
Greenplum Database 4.3 Load Tools for UNIX	A01
Greenplum Database 4.3 Load Tools for Windows	A01
Greenplum Command Center 1.2 Administrator Guide	A01

GoPivotal, Inc. believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS IS." GOPIVOTAL, INC. ("Pivotal") MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.