



Rev: A01

Updated: October 4, 2012

Welcome to Greenplum Database 4.2.2.4

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 of 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 4.2.2.x. For previous versions of the release notes for Greenplum Database, go to [Powerlink](#).

About Greenplum Database 4.2.2.4

Greenplum Database 4.2.2.4 includes new issue resolutions and minor feature enhancements. Please refer to the following sections for more information about this release.

- [Supported Platforms](#)
- [New Functionality in Greenplum Database 4.2.2](#)
- [Upcoming Deprecated Options Post-Greenplum Database 4.2.2](#)
- [Resolved Issues in Greenplum Database 4.2.2.4](#)
- [Known Issues in Greenplum Database 4.2.2.4](#)
- [Upgrading to Greenplum Database 4.2.x.x](#)
- [Greenplum Database Tools Compatibility](#)
- [Greenplum Database Extensions Compatibility](#)
- [Hadoop Distribution Compatibility](#)
- [Greenplum Database 4.2 Documentation](#)

Supported Platforms

Greenplum Database 4.2.2 runs on the following platforms:

- Red Hat Enterprise Linux 64-bit 5.5, 5.6, 5.7, 6.1, and 6.2
- 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, 6.2

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 [Powerlink](#) home page, select:

Home > Support > Interoperability and Product Lifecycle Information > E-Lab Interoperability Navigator, then search for Greenplum Database.

New Functionality in Greenplum Database 4.2.2

The following new functionality is being introduced in this release of Greenplum Database. See the Greenplum Database documentation for more details about each feature. For a complete list of relevant documentation, see [Greenplum Database 4.2 Documentation](#).

Out of Disk Space

Greenplum Database 4.2.2 can proactively monitor and alert users to low levels of remaining disk capacity. You can configure the threshold levels, “warning” and “error”, to receive alerts when disk capacity falls below these levels. You can also define the frequency for receiving alerts via email messages or SNMP traps.

Direct I/O

Greenplum Database 4.2.2 improves NFS-based backups through a new feature called Direct I/O.

Direct I/O allows you to bypass the buffering of memory within the file system cache. When Direct I/O is used for a file, data is transferred directly from the disk to the application buffer, without the use of the file buffer cache. Direct I/O benefits applications by reducing CPU consumption and eliminating the overhead of copying data twice – first between the disk and then from the file buffer cache to the application buffer.

This feature is only available for Red Hat Enterprise Linux platforms.

Location of critical files are now user configurable

SSD and Flash storage are increasingly becoming part of standard server configurations. In Greenplum Database 4.2.2 some system and temp files can be configured to run on a separate filesystem from user data. Heterogenous filesystem support enables placing system and temporary files on SSD/Flash for substantial system performance improvements while still allowing for user data to take advantage of the larger capacity of spinning-disk media.

Inverse Distribution Functions

GPDB 4.2.2 provides support for Inverse Distribution Functions, also known in the area of probability and statistics as quantile functions. Given a percentile, an Inverse Distribution Function returns the specific percentile value for the provided data set. This data set assumes either a continuous or a discrete distribution model. The most commonly used quantile function is `median`, which returns a middle value or interpolated value for your data set. Greenplum Database 4.2.2 contains three new functions that take advantage of the underlying inverse distribution algorithms. These are as follows:

- `percentile_cont()`: assumes a continuous distribution model. It takes a percentile value and sort specification and returns an interpolated value that is the percentile value based on the sort specification.
- `percentile_disc()`: assumes a discrete distribution model. It takes a percentile value and a sort specification and returns an element from the set.
- `median`: assumes a continuous distribution model. It takes a numeric or date time value and returns the middle value or an interpolated value that is the middle value when the values are sorted.

These functions are essentially aggregate functions but are implemented as specialized operators for efficiency in the executor and planner. The `percentile_cont()` and `percentile_disc()` functions use the `WITHIN GROUP` syntax to specify input ordering. You can use the `median` function without the `WITHIN GROUP` syntax. For example:

```
Select median(10);
```

Updatable Cursors

A cursor is a database object that allows users and applications to manipulate data in a row-by-row manner rather than set-based operations. Before Greenplum Database 4.2.2, once cursors were declared, they became read-only. In Greenplum Database 4.2.2, cursors can update on current cursor position after they have been declared.

Upcoming Deprecated Options Post-Greenplum Database 4.2.2

This section provides notice about any options that will be deprecated after Greenplum Database 4.2.2.4 and later. Greenplum provides this information so you can adopt best practices to prepare for these changes:

Table 1 Deprecated options post-4.2.2

Deprecated Option	Description
Named operator =>	Greenplum will not support “=>” as the name of a user-defined operator in future releases of Greenplum Database. Greenplum suggests that you do not use “=>” as a user-defined operator, prior to the next major version upgrade.
INHERITANCE	Greenplum will deprecate the usage of the <code>INHERITS</code> clause in <code>CREATE TABLE</code> DDL in future releases of Greenplum Database.

Resolved Issues in Greenplum Database 4.2.2.4

The table below lists customer-reported issues that are now resolved in Greenplum Database 4.2.2.4

The following table lists the resolved issues in the current release.

Table 2 Resolved Issues in 4.2.2.4

Issue Number	Category	Resolved In	Description
18415	Query Execution	4.2.2.4	When queries meet several simultaneous conditions, incorrect results or SIGSEGV errors may occur. Queries that simultaneously meet several predicate, partitioning, and redistribution requirements can generate a SIGSEGV error or incorrect results. Whether an affected query plan produces a SIGSEGV error or incorrect results is dependent on internal query execution details.
18408	DDL and Utility Statements	4.2.2.3	4.2.1.x and above introduced an OID error in the following circumstance. When a non-superuser tried to drop a type, the order of dropping removed the type, then removed the encoding. When the encoding removal operation tried to find its relevant OID, it could not because it was already dropped (by dropping the <code>pg_type</code> .)
18265	Client Access Methods	4.2.2.2	When the Autocommit option in JDBC was set to false and a rollback occurred in a transaction that has already been aborted, the error "current transaction is aborted, commands ignored until end of transaction block" occurred. JDBC rollbacks are now properly managed to finish the current transaction block as required and no error messages are displayed. The recommended JDBC driver for JDK 1.6 and 1.7 is Postgresql 8.2-512 JDBC4.

Table 2 Resolved Issues in 4.2.2.4

Issue Number	Category	Resolved In	Description
18027	Query Execution	4.2.2.1	<p>Queries that included hashjoins created too many files when certain GUCs were set.</p> <p>GUCS include: gp_workfile_type_hashjoin='BFZ' gp_workfile_compress_algorithm='ZLIB'.</p> <p>When this occurred and the system reached the maximum number of open files per process, the query aborted and it tried to close all files, causing the clean-up code to fail and creating a PANIC. This issue has been resolved in this release.</p>
18024	DDL and Utility Statements	4.2.2.1	<p>Systems could not kill queries using pg_cancel_backend or pg_terminate_backend while they were waiting on locks for unique indexes.</p> <p>This issue has been resolved with this release.</p>
17988	Catalog and Metadata	4.2.2.1	<p>If views contained certain type of decode expressions like "Unexpected interval error", the view users could get errors during de-parsing.</p> <p>This issue has been resolved with this release.</p>
17986	Catalog and Metadata, DDL and Utility Statements	4.2.2.1	<p>If two child partitions had locks from different sessions, the user may have see an error: "could not open relation with OID XXXXX".</p> <p>This issue has been resolved in this release.</p>
17401	None	4.2.2	<p>Kerberos 1.8.1 or higher generated cache credentials that Java could not read.</p> <p>This was a known issue in Java 1.6.0_26 and earlier. Greenplum now distributes JRE 1.6.0_32 instead of JRE 1.6.0_26 along with the database extension packages.</p>
17510	Backup and Restore	4.2.2	<p>gpccrondump failed after deleting the storage unit on Data Domain.</p> <p>When you ran gpccrondump for the first time after deleting a storage unit, the following error message was sometimes displayed:</p> <pre>Dump request was incomplete, not rolling back because -r option was not supplied.</pre>

Table 2 Resolved Issues in 4.2.2.4

Issue Number	Category	Resolved In	Description
17480	4.2.2	None	<p>Multi DD NICs support did not appear in the gpccrondump help menu.</p> <p>During the dump process with Data Domain Boost, you should configure the hostname and IP address of the Data Domain server where you are creating your backups. Prior to 4.2.2, it was only possible to configure all the segment hosts with the same hostname. This meant that even if the Data Domain server had more than one NIC, we were still using only one. This feature allows you to specify several NICs for the Data Domain server. So if you specify 2 hostnames (host1, host2), then half the segments will backup to host1 and the other half to host2. Greenplum does not limit the number of Data Domain hosts.</p> <p>Example:</p> <pre>gpccrondump --ddboost-host ddbboost_hostname1 --ddboost-host ddbboost_hostname2 --ddboost-user ddbboost_user</pre>
17258 17081	4.2.2	gpload	<p>Prior to 4.2.1, gpload jobs whose MATCH_COLUMNS did not contain the target table's distribution key would error out.</p> <p>This restriction has been removed in this release.</p> <p>In addition, gpload supports loading data into randomly distributed table.</p>
16922	4.2.2	Transaction Management	<p>The sub-transaction limit of 100 is removed.</p> <p>You can run any number of sub-transactions as long as you have enough system memory. You may notice slower performance for queries with a large number of sub-transactions.</p>
16824	4.2.1.1	Functions and Languages	<p>Perl-5.12.4 bundled in the gppkg plperl - 1.1 is not compatible with some Perl modules.</p> <p>4.2.2 provides two versions of PL/Perl. PL/Perl 1.1 supports Perl 5.1.24 and PL/Perl 1.2 supports Perl-5.8.8. These are compatible with RHEL 6.x and 5.x respectively. This resolves PL/Perl module compatibility issues.</p>
16766	4.2.2	Upgrade/ Downgrade	<p>Upgrading to 4.2.x release of Greenplum Database accessed unused code.</p> <p>Prior to 4.2.2, the syntax for the CREATE PROTOCOL command permitted the use of a namespace. However, Greenplum Database never recorded nor utilized this namespace. In 4.2.2, user-defined protocols disallow the use of namespaces.</p>
16726 16402	4.2.2	Loader: gpfdist	<p>gpfdist_ssl: cannot upload big files from the ETL server running on different platform (Linux) to the Greenplum cluster running on Solaris.</p> <p>When running gpfdist with the --ssl option on Linux, while the master and segments running on Solaris, the gpfdist returns an error.</p>
16577	4.2.2	Upgrade and Downgrade	<p>gpmigrator --check-only option is documented in the online help and in the Greenplum Database Administrator Guide, Revision A03.</p>

Table 2 Resolved Issues in 4.2.2.4

Issue Number	Category	Resolved In	Description
16520	4.2.2	Documentation	<p>The description of <code>gpkill</code> in the <i>Greenplum Database Administrator Guide</i>, Revision A03, was modified as follows:</p> <p>The <code>gpkill</code> utility checks or terminates a Greenplum process or a list of Greenplum processes. If the process is a critical Greenplum Database process or a system process (not part of Greenplum), <code>gpkill</code> does not terminate it.</p> <p>Users other than the superuser can only use <code>gpkill</code> to terminate their own processes.</p>
16515 13669	4.2.2	Documentation	<p>The section, “Creating Queues with Memory Limit”, in the <i>Greenplum Database Administrator Guide</i>, Revision A03, was modified as follows:</p> <p>The <code>MEMORY_LIMIT</code> setting is determined on a per-segment basis, not on a per-segment-host basis. The information should be edited as follows:</p> <p>“Resource queues that offer a <code>MEMORY_LIMIT</code> setting control the amount of memory for all the queries submitted through the queue. The total memory should not exceed the physical memory available per-segment. Greenplum recommends that you set <code>MEMORY_LIMIT</code> to 90% of memory available on a per-segment basis. For example, if a host has 48 GB of physical memory and 6 segments, then the memory available per segment is 8 GB. You can calculate the recommended <code>MEMORY_LIMIT</code> for a single queue as $0.90 \times 8 = 7.2$ GB. If there are multiple queues created on the system, their total memory limits must also add up to 7.2 GB.”</p>
16428 16118 12655	4.2.2	PL/pgSQL	<p>In certain scenarios when executing PL/pgSQL functions, users will no longer encounter messages.</p> <p>Users previously encountered an error message such as:</p> <pre>ERROR: relation with OID xxxx does not exist</pre>
16425 15886	4.2.2	Query Execution	<p>Queries may fail when run within user-defined functions, even with very high memory statement settings.</p> <p>You may see the error message:</p> <pre>56: ERROR: insufficient memory reserved for statement</pre> <p>Workaround: To turn off memory quotas before running queries within UDFs, set <code>gp_resqueue_memory_policy</code> to <code>none</code>.</p>
16369	4.2.2	Resource Management	<p>Resource-queue deadlock issue.</p> <p>Resource queue deadlocks were encountered if a cursor was associated with a query residing within a function.</p>
16310		Functions and Languages	<p>plpython cannot handle the SPI exception correctly.</p> <p>If plpython encounters the SPI exception it aborts the transaction and discards the changes. The appropriate error message is now displayed.</p>
15769	4.2.2	Documentation	<p>The new sizing recommendations for Data Domain Boost are documented in the <i>Greenplum Database Administrator Guide</i>, Revision A03.</p>

Table 2 Resolved Issues in 4.2.2.4

Issue Number	Category	Resolved In	Description
15719	4.2.2	Documentation	The following incorrect statement has been deleted from the <i>Greenplum Database Administrator Guide, Revision A03</i>. The Greenplum Database Administrator Guide stated that when using the UPDATE and DELETE commands the following restriction applied: “Cannot use STABLE or VOLATILE functions in UPDATE (or DELETE) statement if mirrors are enabled.”
15458	4.2.2	Replication - Master Mirroring	Documentation was not clear about <code>-t</code> and <code>-n</code> options for <code>pg_dump</code>. The description of the <code>-t</code> option was previously: <code>-t, --table=TABLE</code> dump the named table(s) only This description has been changed to <code>-t, --table=TABLE</code> dump the named table(s) only, schema.tablename.

Known Issues in Greenplum Database 4.2.2.4

This section lists the new known issues in Greenplum Database 4.2.2.4. A workaround is provided where applicable.

For known issues discovered in previous releases, including patch releases to Greenplum Database 4.1 or 4.0.x, see the corresponding release notes, available from [Powerlink](#):

Table 3 All Known Issues in 4.2.2.4

Issue	Category	Description
16782/ 16219	Client Access Methods, Dispatch	JDBC statements sometimes fail with <code>Unexpected internal error</code> when unanalyzed tables are accessed. Workaround: Issue an <code>ANALYZE TABLE</code> command on tables accessed via JDBC or issue a <code>setAutoCommit(false)</code> call on the Java Connection object.
17830/ 14640	Backup and Restore	<code>gpdb_restore</code> outputting incorrect non-zero error message. When performing single table restore, <code>gpdb_restore</code> is giving warning messages about non-zero tables but printing out zero rows.
17285	Backup and Restore	NFS backup with <code>gpcrondump -c</code> can fail. In circumstances where you haven't backed up to a local disk before, backups to NFS using <code>gpcrondump</code> with the <code>-c</code> option can fail. On fresh systems where a backup has not been previously invoked there are no dump files to cleanup and the <code>-c</code> flag will have no effect. Workaround: Do not run <code>gpcrondump</code> with the <code>-c</code> option the first time a backup is invoked from a system.
17837	Upgrade/ Downgrade	Major version upgrades internally depend on the <code>gp_toolkit</code> 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 <code>gp_toolkit</code> schema in all affected databases by applying the SQL file found here: <code>\$GPHOME/share/postgresql/gp_toolkit.sql</code> .

Table 3 All Known Issues in 4.2.2.4

Issue	Category	Description
17513	Management Scripts Suite	<p>Running more than one <code>gpfilespace</code> command concurrently with itself to move either temporary files (<code>--movetempfilespace</code>) or transaction files (<code>--movetransfilespace</code>) to a new filespace can in some circumstances cause OID inconsistencies.</p> <p>Workaround: Do not run more than one <code>gpfilespace</code> command concurrently with itself. If an OID inconsistency is introduced <code>gpfilespace --movetempfilespace</code> or <code>gpfilespace --movetransfilespace</code> can be used to revert to the default filespace.</p>
17606	DDL/DML: Partitioning	<p>Unable to add partitions after data type alterations on partitioning key.</p> <p>Invoking <code>ALTER TABLE ALTER COLUMN TYPE</code> on the partitioning key of a partitioned table may render the table unusable.</p> <p>Workaround: Invoking <code>ALTER TABLE ALTER COLUMN TYPE</code> a second time to restore the original type information to the partitioning key.</p>
17780	DDL/DML: Partitioning	<p>ALTER TABLE ADD PARTITION inheritance issue</p> <p>When performing an <code>ALTER TABLE ADD PARTITION</code> 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 <code>ADD PARTITION</code> invocation. For leaf partitions that are already afflicted, the issue can be rectified through use of <code>EXCHANGE PARTITION</code>.</p>
17795	Management Scripts Suite	<p>Under some circumstances, <code>gppkg</code> on SuSe is unable to correctly interpret error messages returned by rpm.</p> <p>On SuSE, <code>gppkg</code> 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.</p>
15692	Backup and Restore	<p>Greenplum Database's implementation of RSA lock box for Data Domain Boost changes backup and restore requirements for customers running SuSE.</p> <p>The 4.2.2 implementation of the RSA lock box for Data Domain Boost login credential encryption only supports customers running on Red Hat Enterprise Linux.</p> <p>Workaround: If you run Greenplum Database on SuSE, use NFS as your backup solution. See the <i>Greenplum Database Administrator Guide</i> for information on setting up a NFS backup.</p>
17604	Security	<p>A Red Hat Enterprise Linux (RHEL) 6.x security configuration file limits the number of processes that can run on gpadmin.</p> <p>RHEL 6.x contains a security file (<code>/etc/security/limits.d/90-nproc.conf</code>) that limits available processes running on gpadmin to 1064.</p> <p>Workaround: Remove this file or increase the processes to 131072.</p>
17422	Backup and Restore	<p><code>gpdump</code> generates incorrect dump results when a non-compressed column was added to a append-only table.</p> <p>In Greenplum Database 4.2.1.0 and above, tables leveraging column compression that have been modified using <code>ALTER TABLE</code> may be in a state where the table-level compression settings conflict with those of individual columns. In such cases, the table schemas are not dumped correctly. This results in a proper dump of the data, but you may see an error when the database attempts to recreate the table during restore.</p>

Table 3 All Known Issues in 4.2.2.4

Issue	Category	Description
16519	Backup and Restore	<p>Limited data restore functionality and/or restore performance issues can occur when restoring tables from a backup where the default backup directory was not used.</p> <p>In order to restore from backup files not located in the default directory you can use the <code>-R</code> 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).</p> <p>Workaround: Define a symbolic link from the default dump directory to the directory used for backup, as shown in the following example:</p> <ol style="list-style-type: none"> 1. Perform a full Database Backup to a specific NFS directory: <pre>\$ gpccrondump -x <db_name> -z -u /backup/DCA-93 -a</pre> 2. Create a file listing the segment servers: <pre>\$ vi /home/gpadmin/segments sdw1 sdw2 sdw3 ...</pre> 3. Remove the relevant date folder from <code>db_dumps</code> directories on the master and segments: <pre>\$ 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'</pre> 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 <code>sdw1</code> was shown here, write a script for the remaining segments: <pre>\$ 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'</pre> 5. Restore from backup files: <pre>\$ gpdbrestore -t 20120619061835 -T <schema.table> -a</pre> 6. Remove the symbolic links <pre>\$ 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'</pre>

Table 3 All Known Issues in 4.2.2.4

Issue	Category	Description
17415	Installer	When you run <code>gppkg -q -info<some gppkg></code>, the system shows the GPDB version 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 <code>/etc/motd</code> file and add the warning message to it. This will send the messages to be redirected to <code>stdout</code> and not <code>stderr</code> . You must encode these warning messages in UTF-8 format.
17227 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, <code>gpfdist</code> fails to bind to socket IPv4, and shows the message “Address already in use”, but binds successfully to socket IPv6.
16556 16321 16129	Management Scripts Suite	gpkill does not run on the Solaris platform. The <code>gpkill</code> 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.
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. <code>gpkill</code> accepts the list of pids but only shows that one of the processes may not be killed.
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 <code>password_hash_algorithm</code> 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 <code>psql</code> , it fails with the appropriate error message.
16064	Backup and Restore	Restoring a compressed dump with the <code>--ddboost</code> option displays incorrect dump parameter information. When using <code>gpdbrstore --ddboost</code> to restore a compressed dump, the restore parameters incorrectly show “Restore compressed dump = Off”. This error occurs even if <code>gpdbrstore</code> passes the <code>--gp-c</code> option to use <code>gunzip</code> for in-line de-compression.
15899	Backup and Restore	When running <code>gpdbrstore</code> with the list (<code>-L</code>) option, external tables do not appear; this has no functional impact on the restore job.

Upgrading to Greenplum Database 4.2.x.x

The upgrade path supported for this release is Greenplum Database 4.1.x.x to Greenplum Database 4.2.x.x. The minimum recommended upgrade path for this release is from Greenplum Database version 4.1.1.5. If you have an earlier version of the database, you must first upgrade to version 4.1.x.

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

- [Planning Your Upgrade](#)
- [Upgrading from 4.2.x.x to 4.2.2](#)
- [Upgrading from 4.1.x.x to 4.2.x.x](#)
- [For Users Running Greenplum Database 4.0.x.x](#)
- [For Users Running Greenplum Database 3.3.x.x](#)
- [Troubleshooting a Failed Upgrade](#)

Planning Your Upgrade

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

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

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

For example:

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

Some of the rules for partitioned tables are different in 4.2 than in previous releases. `gpmigrator` detects partitioned tables that violate these new rules and aborts the upgrade. In most cases, `gpmigrator` will create a repair script you can run to bring your 4.1 Greenplum Database into line with the new rules in 4.2. See [Upgrading Partitioned Tables with Constraints](#) for more details.

Upgrading Partitioned Tables with Constraints

Partition tables with `CHECK`, `PRIMARY KEY`, or `UNIQUE` constraints must be updated prior to upgrading:

- Regular `CHECK`, `PRIMARY KEY`, or `UNIQUE` constraints added by database users usually appear on every sub-table of the partitioned table. Their names may need to be adjusted. If necessary, `gpmigrator` creates a repair script to do this.
- `PRIMARY KEY` and `UNIQUE` constraints on partitioned tables that do not include all the columns of the partition key need to be removed. If needed, `gpmigrator` creates a repair script to do this. Note that the unique index underlying the constraint remains and provides the same protection against duplicate keys as did the constraint. As was previously the case, it is possible for different parts of the partitioned table to contain the same key value.
- If the added constraints appear on some but not all of the sub-tables that make up a partitioned table, they cannot be updated automatically. In this case, you need to either drop the irregular constraints or add the missing constraints. Other causes of irregularity may exist but are rare. Greenplum recommends that you contact support if you encounter any issues with partitioned tables that cannot be resolved automatically with `gpmigrator`.

Upgrading from 4.2.x.x to 4.2.2

These instructions are for users currently running 4.2.x.x. An upgrade from 4.2.x.x to 4.2.2 involves removing the standby master from your Greenplum system (if configured), stopping Greenplum Database, updating the Greenplum Database software binaries, and restarting Greenplum Database.

1. Log in to your Greenplum Database master host as the Greenplum administrative user:

```
$ su - gpadmin
```

If you have a standby master configured, remove it from your Greenplum configuration:

```
$ gpinitstandby -r
```

2. Perform a smart shutdown of your current Greenplum Database 4.2.x.x system (there can be no active connections to the database):

```
$ gpstop
```

3. Run the installer for 4.2.2 on the Greenplum Database master host. When prompted, choose an installation location in the same base directory as your current installation. For example:

```
/usr/local/greenplum-db-4.2.2.x
```

4. Edit the environment of the Greenplum Database superuser (`gpadmin`) and make sure you are sourcing the `greenplum_path.sh` file for the new installation. For example change the following line in `.bashrc` or your chosen profile file:

```
source /usr/local/greenplum-db-4.2.x.x/greenplum_path.sh
```

to:

```
source /usr/local/greenplum-db-4.2.2.x/greenplum_path.sh
```

Or if you are sourcing a symbolic link (`/usr/local/greenplum-db`) in your profile files, update the link to point to the newly installed version. For example:

```
$ rm /usr/local/greenplum-db
$ ln -s /usr/local/greenplum-db-4.2.2.x
  /usr/local/greenplum-db
```

5. Source the environment file you just edited. For example:

```
$ source ~/.bashrc
```

6. Log in as root and install the 4.2.2.x binaries on all of the segment hosts. Tar the Greenplum installation directory, use `gpscp` to copy it to the segment hosts, and use `gpssh` to untar it at the segment hosts. Make sure the `gpadmin` user owns the new installation directory. Also update any symbolic links to point to the new version. For example:

```
$ su -
# gtar -cvf gp.tar /usr/local/greenplum-db-4.2.2.x
# gpscp -f seghostname_file gp.tar =:/
# gpssh -f seghostname_file
=> gtar --directory / -xvf /gp.tar
=> chown -R gpadmin /usr/local/greenplum-db-4.2.2.x
=> rm /gp.tar
=> rm /usr/local/greenplum-db
=> ln -s /usr/local/greenplum-db-4.2.2.x
  /usr/local/greenplum-db
```

7. After all segment hosts have been upgraded, you can log in as the `gpadmin` user and restart your Greenplum Database system:

```
$ su - gpadmin
$ gpstart
```

8. If your original system had a standby master, add the standby master back into your Greenplum configuration:

```
$ gpinitstandby -s standby_hostname
```

9. If you are utilizing Data Domain Boost, you have to re-enter your DD Boost credentials after upgrading from Greenplum Database 4.2.1 to 4.2.2, 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 4.1.x.x to 4.2.x.x

This section describes how you can upgrade from Greenplum Database 4.1.x.x or later to Greenplum Database 4.2.x.x. For users running versions prior to 4.1.x.x of Greenplum Database, see the following:

- [For Users Running Greenplum Database 4.0.x.x](#)
- [For Users Running Greenplum Database 3.3.x.x](#)

Upgrade Checklist

This checklist provides a quick overview of all the steps required for an upgrade from 4.1.x.x to 4.2.x.x. Detailed upgrade instructions are also provided in the [Upgrade Procedure](#) section.

Pre-Upgrade Preparation (on your current system)	
<i>* 4.1.x.x system is up and available</i>	
<input type="checkbox"/>	Log in to your master host as the <code>gpadmin</code> user (your Greenplum superuser).
<input type="checkbox"/>	Check for and recover any failed segments (<code>gpstate</code> , <code>gprecoverseg</code>).
<input type="checkbox"/>	Install the Greenplum Database 4.2.x.x binaries on all Greenplum hosts.
<input type="checkbox"/>	Copy or preserve any additional folders or files (such as backup folders) that you have added in the Greenplum data directories or <code>\$GPHOME</code> directory. Only files or folders strictly related to Greenplum Database operations are preserved by the migration utility.
<input type="checkbox"/>	(Optional) Run <code>VACUUM</code> on all databases, and remove old server log files from <code>pg_log</code> in your master and segment data directories. This is not required, but it will reduce the size of Greenplum Database files to be backed up and migrated.
<input type="checkbox"/>	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.
Upgrade Execution	
<i>* The system will be locked down to all user activity during the upgrade process</i>	
<input type="checkbox"/>	Backup your current databases (<code>gpcrondump</code> or ZFS snapshots) and secure backup files in a location outside of your Greenplum data directories.
<input type="checkbox"/>	Remove the standby master from your system configuration (<code>gpinitstandby -r</code>).
<input type="checkbox"/>	Do a clean shutdown of your current system (<code>gpstop</code>).
<input type="checkbox"/>	Update your environment to source your Greenplum Database 4.2.x.x installation.

<input type="checkbox"/>	Run the upgrade utility (<code>gpmigrator_mirror</code> if you have mirrors, <code>gpmigrator</code> if you do not).
<input type="checkbox"/>	After the upgrade process finishes successfully, your 4.2.x.x system will be up and running.
Post-Upgrade (on your 4.2.x.x system)	
* The 4.2.x.x system is up	
<input type="checkbox"/>	Reinitialize your standby master host (<code>gpinitstandby</code>).
<input type="checkbox"/>	Upgrade <code>gpfdist</code> on all of your ETL hosts by installing the compatible version 4.2.x.x Load Tools package.
<input type="checkbox"/>	Rebuild any custom modules against your 4.2.x.x installation (for example, any shared library files for user-defined functions in <code>\$GPHOME/lib</code>).
<input type="checkbox"/>	Greenplum Database 4.2.x.x introduced the <code>gppkg</code> utility to install Greenplum Database extensions. If you were previously using any PostgreSQL extensions such as <code>pgcrypto</code> , <code>PL/R</code> , <code>PL/Java</code> , <code>PL/Perl</code> , and <code>PostGIS</code> , download the corresponding packages from the EMC Download Center , and install using this utility. See the <i>Greenplum Database Administrator Guide 4.2</i> or later for usage details.
<input type="checkbox"/>	Install the latest Command Center Console and update your environment to point to the latest Command Center binaries. Note that the Greenplum Command Center management tool has replaced Greenplum Performance Monitor. See “ Greenplum Database Tools Compatibility ” on page 21 for detailed compatibility information.
<input type="checkbox"/>	Inform all database users of the completed upgrade. Tell users to update their environment to source the Greenplum Database 4.2.x.x installation (if necessary).

Upgrade Procedure

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



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 4.1.x system)

Perform these steps on your current 4.1.x Greenplum Database system. This procedure is performed from your Greenplum master host and should be executed by the Greenplum superuser (`gadmin`).

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

```
$ su - gadmin
```

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 all existing log files from your segment hosts:

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

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.

Install the Greenplum 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-4.2.x.x-PLATFORM.zip
```

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

```
# /bin/bash greenplum-db-4.2.x.x-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-4.2.x.x), or enter an absolute path to an install location. You must have write permissions to the location you specify.
```

6. The installer will install the Greenplum software and create 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. Create a `hostfile` file that has the machine configured host names and host addresses (interface names) for each host in your Greenplum system (master, standby master and segments). Make sure there are no blank lines or extra spaces. For example, if you have a master, standby master, and three segments with two network interfaces per host, your file would look something like this:

```
mdw
mdw-1
mdw-2
smdw
smdw-1
smdw-2
```

```

sdw1
sdw1-1
sdw1-2
sdw2
sdw2-1
sdw2-2
sdw3
sdw3-1
sdw3-2

```

8. Source the path file from your new 4.2.x.x installation. For example:


```
$ source /usr/local/greenplum-db-4.2.x.x/greenplum_path.sh
```
9. Run the `gpsegininstall` utility referencing the `hostfile` file you just created. Use the `-u` and `-p` options to specify the name and password of your Greenplum administrative user (`gpadmin`). For example:


```
$ gpsegininstall -f hostfile -u gpadmin -p P@$$word
```

Upgrade Execution

During upgrade, all client connections to the master will be locked out.

10. Source the path file from your old 4.1.x.x installation. For example:


```
$ source /usr/local/greenplum-db-4.1.1.5/greenplum_path.sh
```
11. (*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.
12. If your system has a standby master host configured, remove the standby master from your system configuration. For example:


```
$ gpinitstandby -r
```
13. Perform a clean shutdown of your current Greenplum system. For example:


```
$ gpstop
```
14. Source the path file from your new 4.2.x.x installation. For example:


```
$ source /usr/home/greenplum-db-4.2.x.x/greenplum_path.sh
```
15. Update your environment so it is sourcing your new 4.2.x.x installation.
 - a. For example, update the `greenplum-db` symbolic link on the master and standby master to point to the new 4.2.x.x installation directory. For example (as root):


```

# rm -rf /usr/local/greenplum-db
# ln -s /usr/local/greenplum-db-4.2.x.x
  /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-4.2.x.x
   /usr/local/greenplum-db
=> chown -R gpadmin /usr/local/greenplum-db
=> exit
```

- 16.** (*optional but recommended*) Prior to running the migration, perform a pre-upgrade check to verify that your database is healthy by executing the 4.2.x.x version of the `gpmigrator` utility with the `--check-only` option.

- 17.** As `gpadmin`, run the 4.2.x.x 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-4.1.1.5
  /usr/local/greenplum-db-4.2.x.x
```

Note: If the migration does not complete successfully, contact Customer Support (see “[Troubleshooting a Failed Upgrade](#)” on page 20).

- 18.** The migration can take a while to complete. After the migration utility has completed successfully, the Greenplum Database 4.2.x.x 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 4.2.x.x system)

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

```
$ gpinitstandby -s standby_hostname
```

- 20.** If your system uses external tables with `gpfdist`, stop all `gpfdist` processes on your ETL servers and reinstall `gpfdist` using the compatible Greenplum Database 4.2.x.x Load Tools package. Application Packages are available at the [EMC Download Center](#).

- 21.** Rebuild any custom modules against your 4.2.x.x installation (for example, any shared library files for user-defined functions in `$GPHOME/lib`).

- 22.** Greenplum Database 4.2.x.x introduced the `gppkg` utility to install Greenplum Database extensions. If you were previously using any PostgreSQL extensions such as `pgcrypto`, `PL/R`, `PL/Java`, `PL/Perl`, and `PostGIS`, download the corresponding packages from [Powerlink](#), and install using this new utility. See the *Greenplum Database Administrator Guide 4.2* or later for usage details.

- 23.** 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](#).

For Users Running Greenplum Database 4.0.x.x

Users on a release prior to 4.1.x.x cannot upgrade directly to 4.2.x.x.

- 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 [Powerlink](#)).
- Follow the upgrade instructions in these release notes for [Upgrading from 4.1.x.x to 4.2.x.x](#).

For Users Running Greenplum Database 3.3.x.x

Users on a release prior to 4.0.x.x cannot upgrade directly to 4.1.x.

- Upgrade from your current release to 4.0.x.x (follow the upgrade instructions in the latest Greenplum Database 4.0.x.x release notes available on [Powerlink](#)).
- Upgrade the 4.0.x.x release to 4.1.x.x (follow the upgrade instructions in the latest Greenplum Database 4.1.x.x release notes available on [Powerlink](#)).
- Follow the upgrade instructions in these release notes for [Upgrading from 4.1.x.x to 4.2.x.x](#).

Troubleshooting a Failed Upgrade

If you experience issues during the migration process, go to the Support page at [Powerlink](#) 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 Checklist](#).
- Log output from `gpmigrator` and `gpcheckcat` (located in `~/gpAdminLogs`)

Greenplum Database Tools Compatibility

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 4 Greenplum Database Tools Compatibility

Client Package	Description of Contents	Client Version	Server Versions
Greenplum Clients	Greenplum Database Command-Line Interface (psql) Greenplum MapReduce (gmpareduce) ¹	4.2.2.0	4.2.2.0
Greenplum Connectivity	Standard PostgreSQL Database Drivers (ODBC, JDBC) PostgreSQL Client C API (libpq)	4.2.2.0	4.2.2.0
Greenplum Loaders	Greenplum Database Parallel Data Loading Tools (gpfdist, gpload)	4.2.2.0	4.2.2.0
Greenplum Command Center	Greenplum Database management tool.	1.2	4.2.2.0

1. gmpareduce is not available on Windows.

Greenplum Database Extensions Compatibility

Greenplum Database delivers an agile, extensible platform for in-database analytics, leveraging the system's massively parallel architecture. With Release 4.2.x.x, Greenplum enables turn-key in-database analytics via 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 4.2* or later 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.

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 5 Greenplum Database Extensions Compatibility

Greenplum Database Extension	Extension Components	
	Name	Version
PostGIS 1.0 for Greenplum Database 4.2.x.x	PostGIS	1.4.2
	Proj	4.7.0
	Geos	3.2.2
PL/Java 1.0 for Greenplum Database 4.2.x.x	PL/Java	Based on 1.4.0
	Java JDK	1.6.0_26 Update 31
PL/R 1.0 for Greenplum Database 4.2.x.x	PL/R	8.3.0.12
	R	2.13.0
PL/Perl 1.1 and 1.2 for Greenplum Database 4.2.x.x	PL/Perl	Based on PostgreSQL 9.1
	Perl	5.12.4
Pgcrypto 1.0 for Greenplum Database 4.2.x.x	Pgcrypto	Based on PostgreSQL 8.3
Greenplum Hadoop File System	gphdfs	1.0
	gphdfs	1.1
MADlib 1.0 for Greenplum Database 4.2.x.x	MADlib	0.3.0

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 6 Hadoop Extensions Compatibility

Hadoop Distribution	Version
Greenplum HD	Greenplum HD 1.0
	Greenplum HD 1.1
Cloudera	cdh302
Greenplum MR	Greenplum MR 1.0
	Greenplum MR 1.2

Greenplum Database 4.2 Documentation

For the latest Greenplum Database documentation go to [Powerlink](#). Greenplum documentation is provided in PDF format.

Table 7 Greenplum Database Documentation

Title	Part Number	Revision
Greenplum Database 4.2 Installation Guide	300-013-162	A02
Greenplum Database 4.2 Administrator Guide	300-013-163	A03
Greenplum Database 4.2 Client Tools for UNIX	300-013-190	A01
Greenplum Database 4.2 Client Tools for Windows	300-013-191	A01
Greenplum Database 4.2 Connectivity Tools for UNIX	300-013-192	A01
Greenplum Database 4.2 Connectivity Tools for Windows	300-013-193	A01
Greenplum Database 4.2 Load Tools for UNIX	300-013-194	A01
Greenplum Database 4.2 Load Tools for Windows	300-013-195	A01
Greenplum Database 4.2.2.4 Release Notes	N/A	A01
Greenplum Command Center 1.2 Administrator Guide	300-012-844	A01

Copyright © 2012 EMC Corporation. All rights reserved.

EMC 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." EMC CORPORATION 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.

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

For the most up-to-date listing of EMC product names, see EMC Corporation Trademarks on EMC.com

All other trademarks used herein are the property of their respective owners.