
Welcome to Pivotal Greenplum Database 4.3.2

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 4.3.2. For previous versions of the release notes for Greenplum Database, go to [Pivotal Documentation](#) or [EMC Support Zone](#).

About Greenplum Database 4.3.2

Greenplum Database 4.3.2 is a maintenance 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 Feature](#)
- [Downloading Greenplum Database](#)
- [Supported Platforms](#)
- [Resolved Issues in Greenplum Database 4.3.2](#)
- [Known Issues in Greenplum Database 4.3.2](#)
- [Upgrading to Greenplum Database 4.3.2](#)
- [Greenplum Database Tools Compatibility](#)
- [Greenplum Database Extensions Compatibility](#)
- [Hadoop Distribution Compatibility](#)
- [Greenplum Database 4.3.2 Documentation](#)

Product Enhancements

Greenplum Database 4.3.2 includes these enhancements:

- [Managing and Migrating Greenplum Database Objects and Data](#)
- [Managing Spill Files That Are Generated During Query Execution](#)
- [Improved Handling of SQL Queries That Read From External Tables](#)
- [Support of the CSV Format for HDFS External Tables](#)
- [Enhanced Support for Encrypted LDAP Authentication](#)
- [Enhanced Greenplum Command Center Performance](#)

Managing and Migrating Greenplum Database Objects and Data

These Greenplum Database utility enhancements help manage and move database objects and data.

- Greenplum Database `gptransfer` utility has been added. The `gptransfer` utility copies database objects from databases in a source Greenplum Database system to databases in a destination Greenplum Database system.

Note: The `gptransfer` utility is supported only with the IPv4 protocol. The `gptransfer` utility is not supported with the IPv6 protocol.

- The Greenplum Database `gppreload` utility has been added. The `gppreload` utility reloads table data with column data sorted. For tables that were created with the table storage option `APPENDONLY=TRUE` and compression enabled, reloading the data with sorted data can improve table compression. You specify a list of tables to be reloaded the table column to be sorted.
- The `--redirect` and `--report-status-dir` options have been added to the Greenplum Database utility `gpdbrestore`.
 - `--redirect`:** When restoring data from a backup created with the `gpcrondump` utility, you can specify a the `--redirect` option and a database name to restore data to a database that is different than the database specified during back up.
 - `--report-status-dir`:** If you run a the `gpcrondump` operation with the `-u` option to specify the backup directory and that is read only, you can specify `--report-status-dir` so that `gpcrondump` writes the report files to a directory with write privileges. The `--report-status-dir` option specify the absolute path to the directory on the each Greenplum Database host (master and segment hosts) where `gpdbrestore` writes report status files for a restore operation.

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

Managing Spill Files That Are Generated During Query Execution

Greenplum Database creates spill files, also known as workfiles, on disk if it does not have sufficient memory to execute an SQL query in memory.

For Greenplum Database 4.3.2, the default value of 100,000 spill files is sufficient for the majority of queries. However, if you have determined that the query must create more than 100,000 spill files, you can increase the value of the Greenplum Database parameter configuration parameter `gp_workfile_limit_files_per_query`.

For information about managing spill files in Greenplum Database, see the *Greenplum Database Administrator Guide*. For information about the server configuration parameter `gp_workfile_limit_files_per_query`, see the *Greenplum Database Reference Guide*.

Improved Handling of SQL Queries That Read From External Tables

The new Greenplum Database server configuration parameter `readable_external_table_timeout` controls the time that Greenplum Database waits for data from an SQL query that read data from an external table.

For information about Greenplum Database configuration parameters, see the *Greenplum Database Reference Guide*. For information about the external tables, see “Loading and Unloading Data” in the *Greenplum Database Administrator Guide*.

Support of the CSV Format for HDFS External Tables

Greenplum Database external tables enable accessing external files as if they are regular database tables. For external files that contain data in the comma separated values (CSV) format on an Hadoop Distributed File System (HDFS), Greenplum Database supports reading and writing the files with the Greenplum Database `gp_hdfs` protocol.

For information about the external tables, see “Loading and Unloading Data” in the *Greenplum Database Administrator Guide*.

Enhanced Support for Encrypted LDAP Authentication

For Greenplum Database 4.3.2, support for encrypting communication between Greenplum Database and an LDAP server has been enhanced. Greenplum Database 4.3.2 supports LDAP authentication with the TLS/SSL protocol to encrypt communication with an LDAP server:

- LDAP authentication with STARTTLS and TLS protocol
STARTTLS starts with a clear text connection (no encryption) and upgrades it to a secure connection (with encryption).
- LDAP authentication with a secure connection and TLS/SSL (LDAPS)
Greenplum Database uses the TLS or SSL protocol based on the protocol that is used by the LDAP server.

If no protocol is specified, Greenplum Database communicates with the LDAP server with a clear text connection.

For information about encrypting communication between Greenplum Database and an LDAP server, see “Configuring Client Authentication” in the *Greenplum Database Administrator Guide*.

Enhanced Greenplum Command Center Performance

Greenplum Database 4.3.2 includes a new Greenplum Database configuration parameter that improves the performance of Greenplum Command Center when monitoring a Greenplum database.

The Greenplum Database configuration parameter `gpperfmon_log_alert_level` controls which message levels are written to the `gpperfmon` log. The log is used by the Greenplum Command Center.

For information about the Greenplum Database configuration parameter, see the *Greenplum Database Reference Guide*. For information about the Greenplum Command Center, see the *Greenplum Command Center Administrator Guide*.

Changed Feature

The Greenplum Database PL/R extension package has been updated to resolve some PL/R issues. The package version is `pv1.1`.

For information about resolved issues, see “[Resolved Issues in Greenplum Database 4.3.2](#).” For information about Greenplum Database PL/R extension package naming, see “[Package File Naming Convention](#).”

New Server Configuration Parameters

This table lists the server configuration parameters introduced in Greenplum Database 4.3.2.

Table 1 New Server Configuration Parameters in 4.3.2

Parameter Name	Value Range	Default Value	Description	Set Classifications
<code>gp_workfile_limit_files_per_query</code>	integer	100000	Sets the maximum number of temporary spill files (also known as workfiles) allowed per query per segment. Spill files are created when executing a query that requires more memory than it is allocated. The current query is terminated when the limit is exceeded. Set the value to 0 (zero) to allow an unlimited number of spill files.	master session reload
<code>gpperfmon_log_alert_level</code>	none warning error fatal panic	none	Controls which message levels are written to the <code>gpperfmon</code> log. Each level includes all the levels that follow it. The later the level, the fewer messages are sent to the log. Note: If the Greenplum Command Center is installed and is monitoring the database, the default value is <code>warning</code> .	local system restart

Table 1 New Server Configuration Parameters in 4.3.2

Parameter Name	Value Range	Default Value	Description	Set Classifications
readable_external_table_timeout	integer >= 0	0	When an SQL query reads from an external table, the parameter value specifies the amount of time in seconds that Greenplum Database waits before cancelling the query when data stops being returned from the external table. The default value of 0, specifies no time out. Greenplum Database does not cancel the query. If queries that use gpfdist run a long time and then return the error “intermittent network connectivity issues”, you can specify a value for readable_external_table_timeout. If no data is returned by gpfdist for the specified length of time, Greenplum Database cancels the query.	master system reload

Downloading Greenplum Database

The location for downloading Greenplum Database software and documentation has changed.

- Greenplum Database 4.3.x software is available from [Pivotal Network](#).
- Current release Greenplum Database documentation is available from the [Pivotal Documentation](#) site.

Previous release versions of Greenplum Database documentation, as well as other Greenplum Database documents, are available from [Support Zone](#)

Supported Platforms

Greenplum Database 4.3.2 runs on the following platforms:

- Red Hat Enterprise Linux 64-bit 5.5, 5.6, 5.7, 6.1, 6.2, 6.4, and 6.5
- SuSE Linux Enterprise Server 64-bit 10 SP4, 11 SP1, 11 SP2
- 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

Note: Starting with Greenplum Database 4.3.0.0, Solaris is no longer a supported operating system. Please send any questions or comments about the changes to supported platforms to gpdb@pivotal.io

Greenplum Database 4.3.2 supports Data Domain Boost on Red Hat Enterprise Linux.

This table lists the versions of Data Domain Boost SDK and DDOS supported by Greenplum Database 4.3.x.

Table 2 Data Domain Boost Compatibility

Greenplum Database	Data Domain Boost	DDOS
4.3.2.0	2.6.2.0	5.2, 5.3, and 5.4
4.3.1.0	2.6.2.0	5.2, 5.3, and 5.4
4.3.0.0	2.4.2.2	5.0.1.0, 5.1, and 5.2

Greenplum Database support on DCA:

- Greenplum Database 4.3.x, all versions, is supported on DCA V2, and requires DCA software version 2.1.0.0 or greater due to known DCA software issues in older DCA software versions.
- Greenplum Database 4.3.x, all versions, is supported on DCA V1, and requires DCA software version 1.2.2.2 or greater due to known DCA software issues in older DCA software versions.

Resolved Issues in Greenplum Database 4.3.2

The table below lists issues that are now resolved in Greenplum Database 4.3.2.

For issues resolved in prior 4.3 releases, refer to the corresponding release notes available from [Pivotal Network](#).

Table 3 Resolved Issues in 4.3.2

Issue Number	Category	Description
24037	Client Access Methods and Tools	In some cases, when the SQLCancel function was used with the Greenplum Database ODBC driver to cancel the execution of a query, a rollback of the transaction did not occur.
23838	Loaders: Copy/External Tables	When the COPY command copied data from a file and the file contained the character sequence '\r\r\n', a postmaster reset occurred.
23768	Query Execution	In some cases, the clean up of an aborted transaction was not handled correctly and caused a PANIC signal to be issued.
23751	Monitoring: gpperfmon server	A memory leak caused the gpmmon process to consume a large amount of memory and CPU resources.
23735	Languages: PL/Java	In some cases, Greenplum Database did not handle concurrent shared memory operations properly from PL/Java routines. This caused a PANIC signal to be issued.
23708	Backup and Restore	In some cases, running the Greenplum Database gpdbrestore utility with the -T or --table-file option failed with this error: ValueError: need more than 1 value to unpack
23706	Upgrade / Downgrade	The Greenplum Database installer did not support upgrading from a Greenplum Database hotfix.
23647	Vacuum	Performing a VACUUM operation on a partitioned append-optimized table did not correctly reduce the age of the parent table and child tables.
23631	Replication: Segment Mirroring	In some rare cases, the crash recovery of a segment mirror failed due to an inconsistent LSN.
23604	Interconnect	In some cases when a Greenplum Database process was cancelled on the Greenplum Database master, corresponding processes remained running on Greenplum Database segment instances.
23578	gphdfs	For Greenplum Database external tables, the gphdfs protocol that accesses data from files on HDFS now supports the CSV file format. See Support of the CSV Format for HDFS External Tables .
23546	Storage Access Methods	In some cases, a DELETE command that contains a join between an append-optimized table and heap table returned this error: ERROR: tuple already updated by self
23485	Transaction Management	When a single Greenplum Database session ran transactions, temporary files were not removed after the transaction completed. If a the session ran a large number of transactions, the temporary files required a large amount of disk space. This issue has been resolved.
23417	Transaction Management	Some queries against an append-optimized table with compression enabled that contained a column with an unknown data type caused a Greenplum Database SIGSEGV error.

Table 3 Resolved Issues in 4.3.2

Issue Number	Category	Description
23227	Client Access Methods and Tools	For Greenplum Database with GSS Authentication enabled, the database role attribute Valid Until was ignored. The Valid Until parameter is now respected when GSS authentication is enabled.
23222	Client Access Methods and Tools	When Greenplum Database receives a SIGSEGV when running the COPY command, Greenplum Database hangs and continuously log this warning message: copy: unexpected response (3)
23204	Query Execution	In some cases, when a Greenplum Database segment fault occurred during the execution of a PL/R function, PL/R hung and continuously returned the same error message.
23202	Management Scripts: expansion	During the process of adding new hosts, the Greenplum Database expand utility gpexpand did not update the <code>pg_hba.conf</code> files on Greenplum Database hosts with the correct host information.
23174	Languages: R, PLR	In Greenplum Database, a signal handling issue in the R programming language caused a potential for postgres processes to hang when running PL/R functions.
23138	Replication: Segment Mirroring	The gprecoverseg utility failed to recover a Greenplum Database segment that was marked as down when the data directory location for the segment was a symbolic link, and a postgres process was running with the same PID as the PID associated with the down segment.
23067	Loaders: Copy/External Tables	In some cases, when an INSERT FROM SELECT command was run that selected from readable external table and inserted into writable external table, this warning was generated: WARNING select failed on curl_multi_fdset (maxfd 10) (4 - Interrupted system call)
23038	Query Execution	When a query was run that contained a polymorphic, user-defined aggregate function, and Greenplum Database was required to create spill files on disk, the query failed with this error: ERROR: could not determine actual argument type for polymorphic function This issue has been fixed.
23008	Dispatch	In some cases when temporary tables were used, Greenplum Database did not perform the clean up of temporary namespaces properly after a transaction completed and caused a SIGSEGV.
22914	Loaders: Copy/External Tables	When a query joined a heap table with an external table that used the gpfdist protocol, an incorrect plan that returned no results might have been chosen.
22787	Monitoring: gpperfmon server	In some cases, the Greenplum Database gpmmmon process failed. The gpmmmon process is used for Greenplum Database performance monitoring.
22784	Storage Access Methods	After a database expansion, some tables created with APPENDONLY=TRUE and compression enabled consumed much more disk space than before the expansion. To reduce disk space in this situation, the Greenplum Database gppreload utility reloads table data with column data sorted. See Managing and Migrating Greenplum Database Objects and Data .
22706	Management Scripts: master mirroring	The Greenplum Database gpinitstandby utility completed successfully but returned an error when the <code>\$GPHOME/share</code> directory was not writable. Now, the utility returns this warning: Please run <code>gpchk --clean</code> after successful standby initialization.

Table 3 Resolved Issues in 4.3.2

Issue Number	Category	Description
22592	Backup and Restore	When the Greenplum Database <code>gpdbrestore</code> utility could not find files on the Greenplum Database master segment that are used to perform a restore operation, the utility did not return the correct error message.
22413	Query Planner	In some cases, an SQL query that contains the following returned incorrect results: a combination of a median function with other aggregates where the GROUP BY columns are a subset of the table's distribution columns.
22328	Management Scripts	When a Greenplum Database extension package was updated with the Greenplum Database <code>gppkg</code> utility option <code>-u</code> , <code>gppkg</code> did not warn the user that updating a package includes removing all previous versions of the system objects related to the package. Now, the <code>gppkg</code> utility warns the user and lets the user cancel the operation.
22265	Locking, Signals, Processes	Greenplum Database hung due to incorrect lock handling that caused a race condition. The lock handling issue was caused by a compiler optimization.
22205	Replication: Segment Mirroring	In some cases, running the Greenplum Database command <code>gprecoverseg -r</code> to rebalance segment instances failed and caused database catalog corruption.
21916	Interconnect	In some cases when the Greenplum Database query dispatcher encountered connection errors, a postmaster reset occurred.
21867	DDL and Utility Statements	The performance of Greenplum Database truncate operations degraded between restarts of Greenplum Database.
21103	Query Execution	In Greenplum Database, support of subnormal double-precision (float8) numbers differed between Red Hat Enterprise Linux 5 and Red Hat Enterprise Linux 6. For example, the value <code>5e-309</code> was not handled consistently by Greenplum Database on RHEL 5 and RHEL 6. This issue has been resolved.
20600	Query Planner	For some SQL queries that contained a subquery, this error message was returned. ERROR: no parameter found for initplan subquery.
20268	Loaders: Copy/External Tables	In some cases when a COPY command was run, improper memory handling caused a PANIC signal to be issued.
19949	Backup and Restore	If a Greenplum database was backed up and the database name contained upper-case characters, the Greenplum Database <code>gpdbrestore</code> utility did not restore the database with the correct name.
19660	Authentication	Greenplum Database supports LDAP authentication. Previously, an issue in Greenplum Database prevented LDAPS (LDAP over SSL) from functioning. This issue has been resolved. See Enhanced Support for Encrypted LDAP Authentication .
19246	Backup and Restore	When performing a selective restore of a partitioned table from a full backup with the Greenplum Database utility <code>gpdbrestore</code> , the data from leaf partitions are now restored. Previously, when performing a selective restore of a partitioned table, you needed to specify all the individual leaf partitions.
18774	Loaders	External web tables that use IPv6 addresses no longer require a port number when using the default port. In previous releases, a port number was required when using an IPv6 address.
13282	Backup and Restore	The database objects in the <code>gp_toolkit</code> schema were not restored after a database was re-created and then restored with the Greenplum Database <code>gpdbrestore</code> utility. The <code>gp_toolkit</code> objects are now restored when a database is re-created and restored.

Known Issues in Greenplum Database 4.3.2

This section lists the known issues in Greenplum Database 4.3.2. A workaround is provided where applicable.

For known issues discovered in previous 4.3.x releases, see the release notes at from [Pivotal Network](#). For known issues discovered in other previous releases, including patch releases to Greenplum Database 4.2.x, 4.1 or 4.0.x, see the corresponding release notes, available from EMC [Support Zone](#):

Table 4 All Known Issues in 4.3.2

Issue	Category	Description
22798	Management Scripts: expansion, Management Scripts: master mirroring	If it is not possible to use SSH to connect from the Greenplum Database master host to 'localhost' a failure occurs when running the Greenplum Database gpactivestandby or gpexpand utility because of an SSH failure. Workaround: Enable SSH to 'localhost' on the master host to work around this issue.
23646	DML	Running an UPDATE command after a DROP COLUMN and ADD PARTITION command on a partitioned table causes a Greenplum Database segment instance failure.
24031	gphdfs	If a readable external table is created with FORMAT 'CSV' and uses the gphdfs protocol, reading a record fails if the record spans multiple lines and the record is stored in multiple HDFS blocks. Workaround: Remove line separators from within the record so that the record does not span multiple lines.
23924	Backup and Restore	In some cases, performing some operations on an append-optimized table and then performing a full backup with the gpccrondump utility to a Data Domain system with DDBoost fails with this error: ERROR: relation "file_name" does not exist
23824	Authentication	In some cases, LDAP client utility tools cannot be used after running the source command <code>source \$GPHOME/greenplum_path.sh</code> because the LDAP libraries included with Greenplum Database are not compatible with the LDAP client utility tools that are installed with operating system. Workaround: The LDAP tools can be used without running the source command in the environment.
23568	Backup and Restore	When backing up a Greenplum database with the Greenplum Database gpccrondump utility and specifying an NFS directory with the -u option, the gpccrondump utility creates an empty db_dumps directory in the master and segment data directories.
23637	Backup and Restore	When restoring a Greenplum database with the Greenplum Database gpccrondump utility, the utility performs an ANALYZE operation on the entire database. Workaround: When restoring Greenplum database with the Greenplum Database gpccrondump utility, specify the --noanalyze option, and then run the ANALYZE command on the tables that require updated statistics.
23525	Query Planner	Some SQL queries that contain sub-selects fail with this error. ERROR: Failed to locate datatype for paramid 0
22792	Build	Greenplum Database is not certified on Red Hat Enterprise Linux 5.10.

Table 4 All Known Issues in 4.3.2

Issue	Category	Description
22215	Build	<p>Greenplum Database is not certified with these connectivity drivers:</p> <ul style="list-style-type: none"> • Data Direct v 7.022; PowerExchange for Greenplum 9.5.1 • 32-bit Microstrategy ODBC for Greenplum Wire Protocol 6.10.01.80 • Open source ODBC 9.01.0100 and JDBC 9.1.902 Type 4 • SAS/ACCESS 9.3 driver provided with SAS software2
23366	Resource Management	<p>In Greenplum Database 4.2.7.0 and later, the priority of some running queries, cannot be dynamically adjusted with the <code>gp_adjust_priority()</code> function. The attempt to execute this request might silently fail. The return value of the <code>gp_adjust_priority()</code> call indicates success or failure. If 1 is returned, the request was not successfully executed. If a number greater than 1 is returned, the request was successful. If the request fails, the priority of all running queries are unchanged, they remain as they were before the <code>gp_adjust_priority()</code> call.</p>
23492	Backup and Restore,	<p>A backup from a Greenplum Database 4.3.x system that is created with a Greenplum Database back up utility, for example <code>gpcrondump</code>, cannot be restored to a Greenplum Database 4.2.x system with the <code>psql</code> utility or the corresponding restore utility, for example <code>gpdbrestore</code>.</p>
23521	Client Access Methods and Tools	<p>Hadoop YARN based on Hadoop 2.2 or later does not work with Greenplum Database.</p> <p>Workaround: For Hadoop distributions based on Hadoop 2.2 or later that are supported by Greenplum Database, the classpath environment variable and other directory paths defined in <code>\$GPHOME/lib/hadoop/hadoop_env.sh</code> must be to be modified so that the paths point to the appropriate JAR files.</p>
21917	Replication: Segment Mirroring	<p>In some rare cases after the Greenplum Database utility <code>gprecoverseg</code> was run, some append-optimized tables and a persistent table were detected having less data on a mirror segment corresponding to a primary segment.</p>
20453	Query Planner	<p>For SQL queries of either of the following forms:</p> <pre>SELECT columns FROM table WHERE table.column NOT IN subquery; SELECT columns FROM table WHERE table.column = ALL subquery;</pre> <p>tuples that satisfy both of the following conditions are not included in the result set:</p> <ul style="list-style-type: none"> • <code>table.column</code> is NULL. • <code>subquery</code> returns the empty result.
21724	Query Planner	<p>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.</p>
21838	Backup and Restore	<p>When restoring sets of tables with the Greenplum Database utility <code>gpdbrestore</code>, 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.</p> <p>Workaround: Before restoring a set of tables, create the schemas for the tables in the database.</p>
21129	DDL and Utility Statements	<p>SSL is only supported on the master host. It is not supported on segment hosts.</p>

Table 4 All Known Issues in 4.3.2

Issue	Category	Description
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 <code>gpcrondump</code> options <code>--ddboost-host</code> and <code>--ddboost-user</code>.
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 <code>gprecoverseg</code> fails due to inconsistent LSN. Workaround: Stop and restart the database. Then perform a full recovery with <code>gprecoverseg</code> .
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 <code>gpcrondump</code> and <code>gpdbrstore</code> utilities do not handle errors returned by DD Boost or Data Domain correctly. These are two examples: <ul style="list-style-type: none"> • If invalid Data Domain credentials are specified when setting the Data Domain Boost credentials with the <code>gpcrondump</code> utility, the error message does not indicate that invalid credentials were specified. • Restoring a Greenplum database from a Data Domain system with <code>gpdbrstore</code> and the <code>--ddboost</code> option indicates success even though segment failures occurred 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.
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 <i>Greenplum Database Administrator Guide</i> 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, <code>libstdc++</code>) 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.
18713	Catalog and Metadata	Drop language <code>plpgsql</code> cascade results in a loss of <code>gp_toolkit</code> functionality. Workaround: Reinstall <code>gp_toolkit</code> .
18710	Management Scripts Suite	Greenplum Management utilities cannot parse IPv6 IP addresses. Workaround: Always specify IPv6 hostnames rather than IP addresses

Table 4 All Known Issues in 4.3.2

Issue	Category	Description
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	gpdbrstore outputting incorrect non-zero error message. When performing single table restore, gpdbrstore 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.
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.

Table 4 All Known Issues in 4.3.2

Issue	Category	Description
17795	Management Scripts Suite	Under some circumstances, <code>gppkg</code> on SUSE is unable to correctly interpret error messages returned by <code>rpm</code> . On SUSE, <code>gppkg</code> is unable to operate correctly under circumstances that require a non-trivial interpretation of underlying <code>rpm</code> 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 <code>gpadmin</code> . RHEL 6.x contains a security file (<code>/etc/security/limits.d/90-nproc.conf</code>) that limits available processes running on <code>gpadmin</code> to 1064. Workaround: Remove this file or increase the processes to 131072.
17415	Installer	When you run <code>gppkg -q -info<some gppkg></code> , 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 <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.
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 filesystem be done in an environment with an uninterrupted power supply.
17189	Loaders: <code>gpfdist</code>	<code>gpfdist</code> 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.

Table 4 All Known Issues in 4.3.2

Issue	Category	Description
16519	Backup and Restore	<p>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.</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 4 All Known Issues in 4.3.2

Issue	Category	Description
16064	Backup and Restore	Restoring a compressed dump with the <code>--ddbboost</code> option displays incorrect dump parameter information. When using <code>gpdbrestore --ddbboost</code> to restore a compressed dump, the restore parameters incorrectly show "Restore compressed dump = Off". This error occurs even if <code>gpdbrestore</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>gpdbrestore</code> with the <code>list (-L)</code> option, external tables do not appear; this has no functional impact on the restore job.

Upgrading to Greenplum Database 4.3.2

The upgrade path supported for this release is Greenplum Database 4.2.x.x to Greenplum Database 4.3.2. 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 4.3.x to 4.3.2](#)
- [Upgrading from 4.2.x.x to 4.3.2](#)
- [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 4.3.x to 4.3.2

An upgrade from 4.3.x to 4.3.2 involves 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 - gadmin
```

2. Perform a smart shutdown of your current Greenplum Database 4.3.x system (there can be no active connections to the database):

```
$ gpstop
```

3. Run the installer for 4.3.2 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-4.3.2.0
```

4. Edit the environment of the Greenplum Database superuser (gadmin) 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-4.3.0.0/greenplum_path.sh
```

to:

```
source /usr/local/greenplum-db-4.3.2.0/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-4.3.2.0
  /usr/local/greenplum-db
```

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


```
$ source ~/.bashrc
```
6. Run the `gpsegininstall` utility to install the 4.3.2 binaries on all the segment hosts specified in the `hostfile`. For example:


```
$ gpsegininstall -f hostfile
```
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 you are utilizing Data Domain Boost, you have to re-enter your DD Boost credentials after upgrading from Greenplum Database 4.3.x to 4.3.2 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 4.2.x.x to 4.3.2

This section describes how you can upgrade from Greenplum Database 4.2.x.x or later to Greenplum Database 4.3.2. 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 filesystem) 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 be done during a database maintenance period. When the utility checks the database catalog, users cannot access the database.

Migrating a Greenplum Database That Contains Append-Only Tables

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

Append-optimized tables are introduced in Greenplum Database 4.3.0. For information about append-optimized tables see the release notes for Greenplum Database 4.3.0.

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-4.3.2.0-PLATFORM.zip
```
3. Launch the installer using `bash`. For example:


```
# /bin/bash greenplum-db-4.3.2.0-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-4.3.2.0`), 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 4.3.2 installation. For example:


```
$ source /usr/local/greenplum-db-4.3.2.0/greenplum_path.sh
```
8. Run the `gpsegininstall` utility to install the 4.3.2 binaries on all the segment hosts specified in the `hostfile`. For example:


```
$ gpsegininstall -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 4.3.2.0 installation. For example:

```
$ source /usr/home/greenplum-db-4.3.2.0/greenplum_path.sh
```

- 14.** Update the Greenplum Database environment so it is referencing your new 4.3.2 installation.

- a.** For example, update the `greenplum-db` symbolic link on the master and standby master to point to the new 4.3.2 installation directory. For example (as root):

```
# rm -rf /usr/local/greenplum-db
# ln -s /usr/local/greenplum-db-4.3.2.0
  /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-4.3.2.0
  /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 4.3.2 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-4.3.2
```

- 16.** As `gpadmin`, run the 4.3.2 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-4.3.2.0
```

Note: If the migration does not complete successfully, contact Customer Support (see “[Troubleshooting a Failed Upgrade](#)” on page 25).

- 17.** The migration can take a while to complete. After the migration utility has completed successfully, the Greenplum Database 4.3.2 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 4.3.2 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 4.3.2 Load Tools package. Application Packages are available at [Pivotal Network](#).
- 20.** Rebuild any custom modules against your 4.3.2 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 [Pivotal Network](#), and install using this utility. See the *Greenplum Database Utility 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 [Pivotal Network](#).
- 23.** Inform all database users of the completed upgrade. Tell users to update their environment to source the Greenplum Database 4.3.2 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 4.3.2. 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	
<input type="checkbox"/>	Log in to your master host as the <code>gpadmin</code> user (your Greenplum superuser).
<input type="checkbox"/>	(Optional) Run <code>VACUUM</code> on all databases,
<input type="checkbox"/>	(Optional) Remove old server log files from <code>pg_log</code> in your master and segment data directories.
<input type="checkbox"/>	Check for and recover any failed segments (<code>gpstate</code> , <code>gprecoverseg</code>).
<input type="checkbox"/>	Copy or preserve any additional folders or files (such as backup folders).
<input type="checkbox"/>	Install the Greenplum Database 4.3 binaries on all Greenplum hosts.
<input type="checkbox"/>	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	
<input type="checkbox"/>	Backup your current databases.
<input type="checkbox"/>	Remove the standby master (<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 the new Greenplum Database 4.3.2 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 4.3.2 system will be up and running.
Post-Upgrade (on your 4.3 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.
<input type="checkbox"/>	Rebuild any custom modules against your 4.3.2 installation.
<input type="checkbox"/>	Download and install any Greenplum Database extensions.
<input type="checkbox"/>	(Optional) Install the latest Command Center Console and update your environment to point to the latest Command Center binaries.
<input type="checkbox"/>	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 4.3.2.

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 at [Pivotal Documentation](#)).
2. Follow the upgrade instructions in these release notes for [Upgrading from 4.2.x.x to 4.3.2](#).

For Users Running Greenplum Database 4.0.x.x

Users on a release prior to 4.1.x.x cannot upgrade directly to 4.3.2.

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 at [Pivotal Documentation](#)).
3. Follow the upgrade instructions in these release notes for [Upgrading from 4.2.x.x to 4.3.2](#).

For Users Running Greenplum Database 3.3.x.x

Users on a release prior to 4.0.x.x cannot upgrade directly to 4.3.2.

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 at [Pivotal Documentation](#)).

4. Follow the upgrade instructions in these release notes for [Upgrading from 4.2.x.x to 4.3.2](#).

Troubleshooting a Failed Upgrade

If you experience issues during the migration process and have active entitlements for Greenplum Database that were purchased through Pivotal, contact Pivotal Support. Information for contacting Pivotal Support is at <https://support.gopivotal.com>.

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.

Tool packages are available from [Pivotal Network](#).

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 (gmapreduce) 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. gmapreduce is not available on Windows.

The Greenplum Database Client Tools, Load Tools, and Connectivity Tools are supported on the following platforms:

- AIX 5.3L (32-bit)
- AIX 5.3L and AIX 6.1 (64-bit)
- Apple OSX on Intel processors (32-bit)
- HP-UX 11i v3 (B.11.31) Intel Itanium (Client and Load Tools only)
- Red Hat Enterprise Linux i386 (RHEL 5)
- Red Hat Enterprise Linux x86_64 (RHEL 4)
- Red Hat Enterprise Linux x86_64 (RHEL 5 and RHEL 6)
- SUSE Linux Enterprise Server x86_64 (SLES 10 and SLES 11)
- Solaris 10 SPARC32
- Solaris 10 SPARC64
- Solaris 10 i386
- Solaris 10 x86_64
- Solaris 9 SPARC32
- Windows 7 (32-bit and 64-bit)
- Windows Server 2003 R2 (32-bit and 64-bit)
- Windows Server 2008 R2 (64-bit)

- Windows XP (32-bit and 64-bit)

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 [Pivotal Network](#) and install them using the Greenplum Packager Manager (gppkg). See the *Greenplum Database Utility Guide* 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 1.1 for Greenplum Database 4.3.x.x	PL/Java	Based on 1.4.0
	Java JDK	1.6.0_26 Update 31

Table 6 Greenplum Database Extensions Compatibility

Greenplum Database Extension	Extension Components	
	Name	Version
PL/R 1.1 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
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
MADlib 1.5 for Greenplum Database 4.3.x.x	MADlib	Based on MADlib version 1.8

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	2.0.3
PL/Java	1.1
PL/Perl	1.2
PL/R	1.0
Pgcrypto	1.1
MADlib	1.5

Package File Naming Convention

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

pkgname-ver_pvpkg-version_gpdbrel-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 *ossvversion*. 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.

pvpkg-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.

gpdbr-el-OS-version-arch - The compatible Greenplum Database release. For the postGIS package, *gpdbr4.3-rhel15-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

This table lists the 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 4.3.2 Documentation

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

Table 9 Greenplum Database Documentation

Title	Revision
Greenplum Database 4.3.2 Release Notes	A01
Greenplum Database 4.3 Installation Guide	A03
Greenplum Database 4.3 Administrator Guide	A02
Greenplum Database 4.3 Reference Guide	A03
Greenplum Database 4.3 Utility Guide	A03
Greenplum Database 4.3 Client Tools for UNIX	A02
Greenplum Database 4.3 Client Tools for Windows	A02
Greenplum Database 4.3 Connectivity Tools for UNIX	A02
Greenplum Database 4.3 Connectivity Tools for Windows	A02
Greenplum Database 4.3 Load Tools for UNIX	A02
Greenplum Database 4.3 Load Tools for Windows	A02
Greenplum Command Center 1.2.2 Administrator Guide	A01

Copyright © 2015 Pivotal Software, Inc. All rights reserved.

Pivotal Software, 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." PIVOTAL SOFTWARE, 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.

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

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