Arenadata™ Database

Версия - archive_5.21.1-arenadata4

Примечания к выпуску ADB-5.21.1-arenadata4

2021 Arenadata™

Оглавление

1 Новые функции **3**
- 1.1 PXF 5.7.0. Новые функции . . . 4
- 1.2 PXF 5.7.0. Измененные функции . . . 4

2 Бета-Функции **5**

3 Устаревшие функции **6**

4 Ключевые отличия ADB от open-source аналога **7**

5 Поддерживаемые платформы **8**

6 Совместимость дистрибутивов Hadoop **10**
- 6.1 Совместимость дистрибутива Hadoop с помощью PXF . . . 10
- 6.2 Совместимость дистрибутива Hadoop с помощью gphdfs . . . 10

7 Решенные проблемы **11**

8 Известные проблемы и ограничения **15**
- 8.1 Обновление gp_toolkit.gp_bloat_diag . . . 21

Arenadata DB (ADB) – это сервер базы данных с массовой параллельной обработкой (MPP), который поддерживает хранилище данных следующего поколения и крупномасштабную аналитическую обработку. Благодаря автоматическому секционированию данных и выполнению параллельных запросов кластер может работать как один суперкомпьютер базы данных, работающий в десятки или сотни раз быстрее, чем традиционная база данных. ADB поддерживает SQL, параллельную обработку MapReduce и объемы данных от сотен гигабайт до сотен терабайт.

Important: Данная документация содержит информацию о выпуске Arenadata DB 5.21.1-arenadata4. Для предыдущих версий продукта необходимо перейти к соответствующей документации

ADB 5.21.1-arenadata4 – минорная версия с изменением и улучшением некоторых функций и решением проблем.

Important: Контактная информация службы поддержки – e-mail: info@arenadata.io

Глава 1

Новые функции

В бандле **ADB 5.21.1-arenadata4** для **ADCM** добавлены новые функции:

- Команда `COPY ... TO` теперь поддерживает условие *ON SEGMENT* при использовании SQL-запроса для копирования данных. Например, теперь поддерживается следующая команда:

```
COPY (SELECT * FROM testtbl) TO '/tmp/mytst<SEGID>' ON SEGMENT
```

- *ADB 5.20.0* и все будущие выпуски *ADB 5* поставляются только с утилитами *Backup* and *Restore* версии *1.12.1*;
- База данных *ADB* включает в себя новые функции групп ресурсов:
  - При настройке группы ресурсов ADB больше не требуется *MEMORY_LIMIT*. При указании *MEMORY_LIMIT=0* база данных использует глобальный пул общей памяти группы ресурсов для обслуживания запросов, выполняемых в группе.
  - При указании *MEMORY_SPILL_RATIO=0* база данных теперь использует параметр конфигурации сервера *statement_mem* для определения начального объема памяти оператора запроса.

  При совместном использовании *MEMORY_LIMIT=0* и *MEMORY_SPILL_RATIO=0* для настройки группы ресурсов эти новые возможности предоставляют схему управления памятью, аналогичную схеме, предоставляемой очередями ресурсов базы данных **ADB**.
- Включен PXF версии *5.7.0* с новыми и измененными функциями и исправлениями ошибок.
- Для поиска причин проблемы *29917* добавлено расширенное журналирование для персистентных таблиц в процессе восстановления сегментов. Расширенное журналирование включено по умолчанию и добавляет приблизительно 646 байт на каждый файл записи персистентных таблиц. Для того, чтобы отключить расширенное журналирование, установите параметр *debug_filerep_config_print* в значение **false**.
- Улучшение производительности некоторых запросов, производящих сравнение со значением NULL, при использовании планировщика GPORCA. Производительность таких запросов была неоптимальной из-за некорректной оценки кардинальности данных. Оценка кардинальности для указанного типа сравнений была оптимизирована.
- Оптимизация UDP-интерконнекта – сервер возвращает процессы в пул и переиспользует их в том же порядке, что и в предыдущих запросах в данной сессии. Это помогает уменьшить сетевую нагрузку на некоторых облачных платформах.
- Включена обновленная версия MADlib 1.16 с поддержкой глубокого обучения.

1.1 PXF 5.7.0. Новые функции

- Улучшена производительность при чтении файла в одну строку таблицы;
- Можно настроить PXF для автоматического отключения сервера или сброса кучи Java, когда PXF JVM обнаруживает состояние нехватки памяти;
- PXF поддерживает конфигурацию для каждого сервера и для каждого пользователя ADB. С помощью этой функции можно настроить различных пользователей базы данных с разными учетными данными или свойствами доступа к удаленному хранилищу данных в одном определении PXF-сервера.
- Усовершенствован JDBC-коннектор PXF:
  - Улучшена производительность *INSERT* коннектора JDBC для внешней базы данных SQL;
  - Поддержка имперсонации пользователя PXF на уровне сервера JDBC. С помощью этой функции можно использовать коннектор для доступа к внешним базам данных SQL в качестве текущего пользователя базы данных ADB;
  - Улучшена поддержка доступа к сервису Hive через JDBC, сконфигурированного с имперсонацией Hive и без нее, и с использованием проверки идентификации Kerberos.
- JDBC коннектор для PXF получил поддержку пулинга JDBC соединений. Доступны для конфигурации опции переиспользования соединений и таймауты.

1.2 PXF 5.7.0. Измененные функции

- **PXF** теперь предоставляет конфигурацию максимального количества потоков *Tomcat* через файл конфигурации пользователя *$PXF_CONF/conf/pxf-env.sh*. Изменения в значениях по умолчанию сохраняются при обновлении **PXF**. Также **PXF** уменьшил максимальное число потоков *Tomcat* по умолчанию с *300* до *200*.
- В **PXF** реализована поддержка именованных JDBC-запросов записанных как с использованием завершающего символа точки с запятой (;), так и без него. В предыдущих версиях требовалось не указывать завершающий символ точки с запятой.
- В поставку **PXF** включена обновленная версия драйвера PostgreSQL JDBC (postgresql-42.2.5.jar).
- С этой версии JDBC-коннектор **PXF** при использовании опций партицирования PARTITION_BY, RANGE и INTERVAL всегда возвращает полный набор данных. В предыдущих версиях запрашивались только данные, соответствующие RANGE и не содержащие NULL или конец интервала RANGE.
- С этой версии **PXF** использует исключительно JSON-парсер *com.fasterxml.jackson*. Ранее использовались одновременно библиотеки *fasterxml* и *org.codehaus.jackson*.

Глава 2

Бета-Функции

Поскольку база данных **ADB** основана на open-source аналоге, она включает в себя несколько экспериментальных функций, позволяющих заинтересованным разработчикам экспериментировать с их использованием в системах разработки. Обратная связь помогает стимулировать разработку данных функций, и они могут поддерживаться в будущих версиях продукта.

Important: Бета-Функции не поддерживаются для развертывания в продуктивной среде

В **ADB 5.21.1-arenadata4** входят следующие экспериментальные функции:

- API-интерфейс плагина хранения для утилит **gpbackup** и **gprestore**. Партнеры, клиенты и разработчики OSS могут совместно использовать плагины при помощи **gpbackup** и **gprestore**;
- **Recursive WITH Queries** (общие выражения таблицы).
- **Resource groups** остаются бета-функцией только на платформе **SuSE 11** из-за ограниченной функциональности **cgroups** в ядре. **SuSE 12** решает проблемы **cgroup** в **Linux**, вызывающие снижение производительности при включении групп ресурсов базы данных **ADB**.
- Коннекторы **Platform Extension** (**PXF**) для записи Parquet-данных.

Глава 3

Устаревшие функции

Следующие функции базы данных **ADB** устарели:

- Утилита **gptransfer** будет удалена в следующем основном выпуске. Утилита копирует объекты между системами базы данных;
- Протокол внешних таблиц **gphdfs** будет удален в следующем основном выпуске. Рекомендуется использовать протокол внешней таблицы **pxf** (Platform Extension Framework) для доступа к данным, хранящимся во внешней файловой системе **Hadoop**.

Глава 4

Ключевые отличия ADB от open-source аналога

ADB 5.x имеет ряд ключевых отличий от open-source аналога:

- Установка продукта с помощью **ADCM**;
- Поддержка сжатия **ZStandard**;
- Интеграция с бандлом мониторинга **ADCM**;
- Поддержка расширенного двустороннего обращения к источникам и приемникам данных через **JDBC**.

Глава 5

Поддерживаемые платформы

ADB 5.21.1-arenadata4 работает на следующих платформах:

- Red Hat Enterprise Linux 64-bit 7.x;
- Red Hat Enterprise Linux 64-bit 6.x;
- SuSE Linux Enterprise Server 64-bit 12 SP2 и SP3 с версией ядра выше 4.4.73-5;
- SuSE Linux Enterprise Server 64-bit 11 SP4;
- CentOS 64-bit 7.x;
- CentOS 64-bit 6.x.

Important: Для поддерживаемых операционных систем Linux база данных ADB поддерживается на системных хостах с использованием процессоров AMD или Intel на основе архитектуры x86-64. Arenadata рекомендует использовать однородный набор аппаратных средств (системных хостов) в системе базы данных ADB

Значительное ухудшение производительности базы данных **ADB** наблюдалось при включении управления нагрузкой на основе групп ресурсов в системах **Red Hat 6.x**, **CentOS 6.x** и **SuSE 11**. Данная проблема вызвана ошибкой ядра **Linux cgroup**. Ошибка ядра исправлена в системах **CentOS 7.x** и **Red Hat 7.x**.

Если используется **Red Hat 6** и производительность с группами ресурсов приемлема, необходимо обновить ядро до версии *2.6.32-696* или выше, чтобы воспользоваться исправлениями для групп. **SuSE 11** не имеет версии ядра, разрешающего эту проблему; группы ресурсов по-прежнему считаются экспериментальной функцией на данной платформе.

ADB на **SuSE 12** поддерживает группы ресурсов для использования в продуктивной среде. **SuSE 12** разрешает проблемы ядра **Linux cgroup**, которые вызывают ухудшение производительности при включенных группах ресурсов в **ADB**.

Для базы данных **ADB**, установленной на **Red Hat Enterprise Linux 7.x** или **CentOS** версии *7.x - 7.3*, проблема с ОС может привести к зависанию **ADB**, на которой выполняется большая рабочая нагрузка. Данная проблема вызвана ошибками ядра **Linux**, которые исправлены в **RHEL 7.3** и **CentOS 7.3**.

База данных **ADB** на системах **SuSE Linux Enterprise** не поддерживает следующие функции:

- Язык PL/Perl;
- Утилиту gpmapreduce;
- Расширение языка PL/Container;
- Платформу Platform Extension Framework (PXF).

ADB 5.21.1-arenadata4 поддерживает следующие версии *Java*:

- *8.xxx*;
- *7.xxx*.

ПО **ADB 5.21.1-arenadata4**, работающее в системах **Linux**, использует **OpenSSL 1.0.2l** (с **FIPS 2.0.16**), **cURL 7.54**, **OpenLDAP 2.4.44** и **Python 2.7.12**. Клиентское ПО базы данных **ADB**, работающее в системах **Windows** и **AIX**, использует **OpenSSL 0.9.8zg**.

ADB 5.21.1-arenadata4 не поддерживает драйвер **ODBC** для **Cognos Analytics V11**. Подключение к ПО **IBM Cognos** с драйвером **ODBC** не поддерживается. База данных **ADB** поддерживает подключение к ПО **IBM Cognos** с помощью драйвера **DataDirect JDBC**.

Следующие примечания описывают поддержку платформы для **ADB**:

- Для запуска базы данных **ADB** поддерживается только файловая система **XFS**. Все остальные файловые системы не поддерживаются;
- База данных **ADB** поддерживается на всех подуктовых серверах *1U* и *2U* с локальным хранилищем. Оборудование специального назначения, не являющееся продуктовым, может поддерживаться по собственному усмотрению на основе общего сходства оборудования с продуктовыми серверами;
- База данных **ADB** поддерживается в сетевом или общем хранилище в том случае, если общее хранилище представлено в виде блочного устройства на серверах, на которых запущена база данных **ADB**, и файловая система **XFS** подключена к блочному устройству. Сетевые файловые системы не поддерживаются. При использовании сетевого или общего хранилища зеркальное отображение базы данных **ADB** должно использоваться так же, как и для локального хранилища, при этом никакие изменения не могут быть внесены в схему зеркалирования или схему восстановления сегментов. Другие функции общего хранилища, такие как де-дублирование и/или репликация, напрямую не поддерживаются базой данных **ADB**, но могут использоваться с поддержкой поставщика хранилища, если они не мешают ожидаемой работе **ADB**;
- База данных **ADB** поддерживается при работе в виртуализированных системах, если хранилище представлено в виде блочных устройств, а файловая система **XFS** подключена к хранилищу каталогов сегментов;
- Для конфигурации системы требуется минимум 10-гигабитная сеть;
- Для **Red Hat Enterprise Linux 7.2** или **CentOS 7.2** параметр *systemd*, установленный по умолчанию на *RemoveIPC=yes* удаляет подключения *IPC* при выходе несистемных пользователей. Это приводит к сбою утилиты **gpinitsystem** базы данных **ADB**.

Глава 6

Совместимость дистрибутивов Hadoop

ADB обеспечивает доступ к **HDFS** с помощью **gphdfs** и **PXF**:

- *Совместимость дистрибутива Hadoop с помощью PXF*
- *Совместимость дистрибутива Hadoop с помощью gphdfs*

6.1 Совместимость дистрибутива Hadoop с помощью PXF

PXF поддерживает **Arenadata Hadoop**, **Cloudera**, **Hortonworks Data Platform** и базовые дистрибутивы **Apache Hadoop**.

При необходимости доступа к данным формата *JSON* **PXF** требует версию дистрибутива **Cloudera Hadoop 5.8** или выше, **Arenadata Hadoop 1.4.1** или выше.

6.2 Совместимость дистрибутива Hadoop с помощью gphdfs

Поддерживаемые дистрибутивы **Hadoop**, совместимые с **gphdfs** перечислены в таблице:

Таблица6.1.: Поддерживаемые дистрибутивы Hadoop, совместимые с gphdfs

Дистрибутив Hadoop	Версия	gp_hadoop_target_version
Cloudera	CDH 5.x	cdh
Hortonworks Data Platform	HDP 2.x	hdp
MapR	MapR 4.x, MapR 5.x	mpr
Apache Hadoop	2.x	hadoop
Arenadata Hadoop	1.4.1 и выше	hdp

Important: MapR требует MapR client

Глава 7

Решенные проблемы

В разделе перечислены проблемы в **ADB 5.21.1-arenadata4**, их решения и результат.

- MPP-29348 – PL/Container

  Проблема совместимости библиотек сделала невозможным совместную установку PL/Container и PostGIS.

  Проблема решена.

- 29563 – gpperfmon

Исправлена проблема, из-за которой *gpperfmon_install* записывал пароль с открытым текстом в окно ввода. Код изменен с целью замены текста пароля звездочками (********).

- 29728 – Query Optimizer

Исправлена проблема в алгебризере Query Optimizer, где оптимизатор неправильно нормализовал подзапросы в запросе со сложным целевым списком *GROUP BY*. Это включало статистическую функцию, на которую есть ссылка в условии *HAVING* или в списке заданий *GROUP BY*. В некоторых случаях такой запрос возвращался к Postgres Planner.

Оптимизатор запросов теперь правильно поддерживает подзапросы с целевыми списками *GROUP BY*, которые содержат внешние ссылки, даже в других (вложенных) подзапросах.

- 29881 – Server COPY

Внутренняя функция, используемая для выполнения оператора *COPY*, не проверяла должным образом состояние соединения. Это могло привести к ситуации, когда сбой сегмента приводит к сбою оператора *COPY*, и оператор не может быть отменен с помощью функции *pg_terminate_backend ()*. Проблема решена.

- 29895 – Query Optimizer

GPORCA потратил много времени на оптимизацию запроса с помощью левых соединений нескольких таблиц с большим количеством столбцов.

Проблема решена. GPORCA теперь исключает несистемные столбцы, на которые нет ссылок в запросе, из списка выходных столбцов, что значительно сокращает время оптимизации запросов этого типа.

- 29905 – Performance

Для запросов, которые сравнивают столбец с большим массивом констант с помощью условия *IN*, GPORCA не создавала эффективный план, использующий индексы, и такие запросы выполнялись дольше.

Проблема решена. GPORCA теперь генерирует статистику для предикатов, имеющих большой *IN*-список, и создает более эффективные альтернативы индексного плана во время оптимизации.

- 29906 – GPORCA

GPORCA не использовала bitmap-индекс для определенных запросов, когда *LIMIT* увеличен с *1* до *2*.

Проблема решена путем повышения стоимости сканирования bitmap-таблиц в GPORCA.

- 29941 – Analyze

При запуске *ANALYZE* для большой таблицы может возникнуть *PANIC*, если значение *SET STATISTICS* для столбца было установлено на высокое значение. Проблема вызвана целочисленным переполнением при выборке строк таблицы.

Проблема исправлена. *ANALYZE* теперь может выбирать необходимые строки без генерации целочисленного переполнения.

- 164636728, 29767 – Storage: Catalog and Metadata

Если в таблице есть как *PRIMARY KEY*, так и условие *DISTRIBUTED BY*, столбцы *DISTRIBUTED BY* должны быть равны первичному ключу или его левому подмножеству. Однако это правило можно нарушить, добавив первичный ключ после создания таблицы.

Проблема решена. Правило теперь применяется при изменении таблицы для добавления первичного ключа.

- 166225997 – gpcrondump

В некоторых случаях *gpcrondump* не регистрировал ошибку, если сегмент был недоступен из-за сбоя сети.

Проблема решена.

- 166294234 – gpcheck

Утилита *gpcheck* сообщала об ошибке, когда значения, рекомендованные в документации по базе данных ADB, были использованы для параметра *net.ipv4.ip_local_port_range*.

Проблема решена. *gpcheck* сообщает об ошибке для этого параметра, только если найдет значения, которые не находятся в допустимых или рекомендуемых пределах.

- 29563 – gpperfmon_install

При запуске утилиты *gpperfmon_install* с параметрами *–password* и *–verbose* пароль передается в виде открытого текста на стандартный вывод и в журнал.

Проблема решена. В выводе команды и сообщениях журнала пароль теперь заменяется звездочками.

- 166807453 – GPORCA

Решена проблема генерации GPORCA некорректного результата для агрегирующего запроса, содержащего связанный подзапрос в предложении SELECT.

- 29914/29956 – Планировщик

Запросы, использующие предложение DISTINCT, могли быть причиной аварийного завешения сегмента или некорректного результата в случае применения планировщика Postgres, когда для *gp_segments_for_planner* было установлено большое значение. Проблема происходила на motion-узле, потому что планировщик некорректно анализировал несоответствующий кортеж при определении, присутствуют ли элементы хэш-выражения подплана также и в списке целей. Эта проблема была решена исправлением ошибки разбора выражения в планировщике Postgres.

- 29943 – GPORCA

Запросы, содержащие предложение UNION для большого числа потомков, могли прерываться и становиться причиной аварийного восстановления на мастере. Проблема была решена удаление ненужной проверки в GPORCA, которая увеличивала время, необходимое для оптимизации таких запросов.

- 29975 – GPORCA, Planner

Могло происходить переключение с GPORCA на планировщик Postgres для запросов, содержащих большое количество соединений, даже в тех случаях, когда планировщик Postgres не мог выполнить подобные запросы. Теперь GPORCA корректно обрабатывает такие запросы без переключения на планировщик Postgres.

- 165690728 – Оптимизатор запросов

В некоторых случаях GPORCA генерировал неоптимальный план, когда условие соединения принимало значение NULL логического типа. Проблема была исправлена. GPORCA генерирует оптимальный план для таких запросов.

- 30011 – ANALYZE

Исправлена ошибка при выполнении ANALYZE для запроса, обращающегося к таблице с большим количеством числовых столбцов. В сообщении об ошибке указывалось, что целевые списки не могут содержать более 1664 элементов. С этой версии ANALYZE возможно применить для анализа запросов, обращающихся к таблицам с более чем 830 числовыми столбцами.

- 29976 – GPORCA

Иногда GPORCA мог генерировать критическую ошибку в процессе упрощения предикатов в тех случаях, когда запрос содержит предикаты равенства между типами данных *char* и *varchar*. Ошибка была исправлена подавлением шага упрощения в случаях, когда это могло привести к ошибке.

- 29972 – Оптимизатор запросов

Исправлена ошибка, из-за которой план запроса, сгенерированный GPORCA, содержащий соединение с выражением в форме `f(a) = b`, не мог быть завершен.

- 29933 – Segment Mirroring

В версии 5.21 исправлена взаимоблокировка между *gprecoverseg* и сервисом передачи файлов (the file transfer service – FTS) в момент добавления или удаления утилитой *gprecoverseg* зеркала сегмента.

- 29580 – Управление и мониторинг

Исправлена проблема некорректного создания дополнительного файла в процессе ротации журналов. Метка времени в имени создаваемого файла была позже, чем текущее время и такой файл не использовался.

- 8094 (29999) – UDP Interconnect

Исправлена ошибка связанная с номерами пар портов отправки/получения, открытых в заданный период времени, проявившая себя на некоторых облачных платформах. С этой версии сервер сохраняет тот же порядок, когда использует повторно процессы из пула.

- 8092, 8095 (29999) – TCP Interconnect

Исправлена проблема, при которой запрос мог быть прерван с ошибкой в случае, когда один бэкэнд-процесс не мог занять порт из-за некорректной обработки ошибок в motion-модуле. В этой версии ненужные операции занятия порта были удалены и сервер корректно обрабатывает данный сценарий.

- 7946 – Оптимизатор запросов

В некоторых случаях GPORCA генерировал некорректный результат в случае включения целевого списка в связанный подзапрос. Проблема была решена, GPORCA возвращает корректный результат для таких ситуаций.

- 26225 – gpcheckcat

Утилита *gpcheckcat* завершалась с ошибкой при генерации итогового отчета, содержащего для сегмента потерянные записи таблицы TOAST не ссылающиеся ни на одну таблицу. Исправлено. Для таких записей выводится строка `N/A`.

29934 – Оптимизатор запросов

Выполнение запроса, содержащего в предложении `IN` подзапрос `SELECT DISTINCT`, для которого план был сгенерирован GPORCA, приводило к критической ошибке **ADB**. Данная проблема была исправлена.

167267884, 167267868 – gpcrondump

Увеличено максимальное значение конфигурационного параметра *extra_float_digits* до 3. gpcrondump использует при выполнении такое значение для данного параметра. Это исправляет проблему, при которой для столбцов типа *float4*, используемых в качестве ключей дистрибуции, не использовалась достаточная точность при резевервном копировании, позволяющая значениям соответствовать исходной политике дистрибуции после восстановления из резервной копии.

167300801 – Оптимизатор запросов

В некоторых случаях GPORCA выделял память для объектов, которые никогда не использовались. Теперь память для этих объектов выделяется только в тех случаях, когда это необходимо. Это позволило существенно сократить потребление памяти для некоторых запросов.

165835701 – Оптимизатор запросов

GPORCA изначально был написан с возможностью оптимизации запросов используя многопоточность. Однако, эта возможность не использовалась. Код, связанный с многопоточной обработкой был удален из GPORCA. В результате была улучшена производительность запросов и их анализа.

29887 – Сервер

Исполнитель запросов передавал некорректный адрес для массива выполняющихся транзакций. Проблеме могли быть подвержены расширенные запросы, такие как `DECLARE ... CURSOR`, а также `RETURN` и `EXECUTE` выражения PL/pgSQL. Проблема была решена.

29752 – Сервер

Вставка в партицированную append-optimized таблицу завершалась с ошибкой в случае выполнения одновременно с операцией VACUUM для таблицы. Ошибка происходила потому что при вставке не блокировались дочерние партиции. Это было исправлено. Теперь при вставке блокируются дочерние таблицы.

Глава 8

Известные проблемы и ограничения

ADB 5.x имеет следующее ограничение:

- Пакеты **ADB 4.3.x** не совместимы с **ADB 5.x**.

Далее перечислены основные известные проблемы в **ADB 5.x**:

- **29917 – Segment Mirroring**

Ошибка `read beyond eof` появляется для некоторых персистентных таблиц в процессе полного восстановления. Источник данной проблемы пока не определен. В версии 5.21 добавлено расширенное отладночное журналирование для выявления причин проблемы. Отладочное журналирвоание включено по умолчанию и добавляет приблизительно 646 байт на каждый файл элемента персистентных таблиц. Для того чтобы отключить расширенное журналирование, установите конфигурационный параметр *debug_filerep_config_print* в *false*.

- **MPP-29861 – gpstart**

Если каталог данных сегмента становится недоступным или содержимое каталога данных удаляется, сегмент переходит в автономный режим и помечается как отключенный в *gp_segment_configuration*. Даже при наличии временных файлов или файлов транзакций в отдельном файловом пространстве или при остановке и запуске базы данных **gpstart** может завершиться с ошибкой:

```
20190404:17:26:44:025089 gpstart:mdw:gpadmin-[ERROR]:-Multiple OIDs found in flat files
20190404:17:26:44:025089 gpstart:mdw:gpadmin-[ERROR]:-Filespaces are inconsistent. Abort Greenplum Database␣
↪start.
```

Ошибка указывает на то, что в каталоге сегментов отсутствуют два файла: *gp_teditional_files_filespace* и *gp_transaction_files_filespace*. Утилита **gpstart** сообщает об ошибке, даже если сегменты уже были отмечены.

Решение: вручную создать отсутствующие файлы в каталоге сегмента, используя соответствующий *dbid* сегмента, каталог сегмента файла транзакции и *oid* файлового пространства.

- **164671144 – gpssh-exkeys**

Утилита **gpssh-exkeys** использует библиотеку **Paramiko SSH** для **Python**, которая зависит от библиотеки **Python Cryptography Toolkit** (**PyCrypto**). В некоторых версиях **PyCrypto** были обнаружены следующие уязвимости безопасности:

- https://nvd.nist.gov/vuln/detail/CVE-2018-6594 – *lib/Crypto/PublicKey/ElGamal.py* в PyCrypto через *2.6.1* генерирует слабые параметры *ElGamalkeyparameters*, что позволяет злоумышленникам получать конфиденциальную информацию путем чтения данных зашифрованных данных (т.е. он не имеет семантической защиты перед лицом атаки только зашифрованного текста). Предположение Decisional

Diffie-Hellman (DDH) не выполняется для реализации *ElGamal*. Paramiko не импортирует этот алгоритм, поэтому база данных ADB не подвержена этой уязвимости.

- https://nvd.nist.gov/vuln/detail/CVE-2013-7459 – переполнение буфера на основе кучи в функции *ALGnew* в *block_templace.c* в Python Cryptography Toolkit (он же pycrypto) позволяет удаленным злоумышленникам выполнять произвольный код, что демонстрируется специально созданным параметром *iv* для *cryptmsg.py*. Эта ошибка была введена в PyCrypto 2.6. База данных ADB имеет PyCrypto 2.0.1 и не подвержена этой уязвимости.
- https://nvd.nist.gov/vuln/detail/CVE-2013-1445 – функция *Crypto.Random.atfork* в PyCrypto до *2.6.1* неправильно повторно заполняет генератор псевдослучайных чисел (PRNG) перед тем, как дочерний процесс получает к нему доступ, что облегчает злоумышленникам, зависящим от контекста, получение конфиденциальной информации путем использования состояние гонки, при котором создается дочерний процесс и получает доступ к PRNG в течение того же периода ограничения скорости, что и другой процесс. В версии *1.7.6-9* Paramiko, используемой в базе данных ADB 4.x и 5.x, появился обходной путь для этой ошибки.
- https://nvd.nist.gov/vuln/detail/CVE-2012-2417 – PyCrypto до *2.6* не генерирует соответствующие простые числа при использовании схемы *ElGamal* для создания ключа, что уменьшает пространство сигнатур или пространство открытых ключей и облегчает злоумышленникам проведение атак методом “грубой силы” для получения закрытого ключа. Paramiko не импортирует этот алгоритм, поэтому база данных ADB не подвержена этой уязвимости.

Посредством тестирования и исследования **Arenadata** определила, что эти уязвимости не влияют на **ADB**, и никаких действий не требуется для существующих версий базы данных. Однако в библиотеке **PyCrypto** могут быть другие неопознанные уязвимости, и пользователи, которые устанавливают более позднюю версию **PyCrypto**, могут быть им подвержены.

Библиотека **PyCrypto** будет удалена из базы данных **ADB 6.0**.

Обходной путь: администраторы могут установить SSH без пароля между хостами в кластере базы данных без использования утилиты **gpssh-exkeys**. Это необходимо сделать до инициализации системы **ADB**:

1. Создать ключ SSH для пользователя *gpadmin* на каждом хосте. Например, войти на каждый хост как *gpadmin* и использовать команду *ssh-keygen* для генерации ключа SSH. Не вводить пароль.
2. Добавить открытый ключ для каждого хоста в кластере для каждого другого хоста в кластере. Например, использовать команду *ssh-copy-id* с каждого хоста, чтобы скопировать открытый ключ на каждый другой хост в кластере.

При добавлении новых хостов в систему базы данных **ADB** необходимо создать новый ключ SSH для каждого нового хоста и обменяться ключами между существующими хостами и новыми.

- **165434975 – search_path**

Известной уязвимости PostgreSQL (https://nvd.nist.gov/vuln/detail/CVE-2018-1058) также подвержена и **ADB**. Проблема связана со схемой по умолчанию *public* и тем, как **ADB** использует настройку *search_path*. Возможность создавать объекты с одинаковым именем в различных схемах вкупе с тем, как **ADB** осуществялет поиск объектов, предоставляет возможность одному пользователю менять поведение запросов других пользователей. Например, злоумышленник может создать функцию, выполнение которой супер-пользователем приведет к повышению привилегий злоумышленника.

Существуют методы защиты от данной уязвимости. Смотрите статью “A Guide to CVE-2018-1058: Protect Your Search Path” в PostgreSQL wiki для получения полной информации о проблеме и способах защиты Ваших данных.

- **29766 – VACUUM**

Длительный запрос каталога может блокировать операции *VACUUM* в системном каталоге до завершения или отмены запроса. Этот тип блокировки не может наблюдаться с помощью **pg_locks**, а сама операция *VACUUM* не может быть отменена до завершения или отмены запроса.

- **29699 – ANALYZE**

В базе данных **ADB** версии *5.15.1-arenadata4* и ниже команда *ANALYZE* может возвращать ошибку *"target lists can have at most 1664 entries"* при выполнении операции *ANALYZE* в таблице с большим количеством столбцов (более *800*). Ошибка возникает по причине того, что in-memory sample таблица, созданная *ANALYZE*, требует дополнительного столбца для указания, является ли колонка NULL или усеченной для каждого анализируемого столбца с переменной длины (например, varchar, text и bpchar, numeric, arrays и geometric datatype). Ошибка возвращается при попытке создать sample таблицу в том случае, когда количество столбцов (столбцов таблицы и колонка индикатора) превышает их максимально допустимое количество.

В базе данных **ADB** версии *5.17.0-arenadata4* и выше ANALYZE sample таблицы не требует дополнительного столбца для текстовых колонок переменной длины, таких как varchar, text и bpchar.

Решение: для сбора статистики по таблице выполнить операции *ANALYZE* по 2 или более наборам колонок таблицы.

- **29674 – VACUUM**

Выполнение параллельных операций *VACUUM* над таблицей каталога, такой как *pg_class*, *gp_relation_node* или *pg_type*, приводит к безвыходному положению и блокирует соединения с базой данных.

Решение: Избегать выполнения параллельных операций *VACUUM* над таблицами каталога и пользовательскими таблицами.

- **29139 – DML**

При первичной проверке leaf-партиций в append-optimized partitioned таблице **ADB** выполняет блокировку *ROW EXCLUSIVE* на все партиции таблицы при вставке данных непосредственно в одну из них. В дальнейшем, пока информация о валидации остается в памяти, вставка данных в leaf-партицию не сопровождается блокировкой других партиций.

Проблема не возникает для heap-storage partitioned таблиц.

- **29523 – gptoolkit**

Обновление с младших версий не обновляет базу данных *template0*, и в некоторых случаях использование этих представлений в схеме *gp_toolkit* может вызвать проблемы при создании базы данных с использованием *template0* в качестве шаблона после обновления до версии **ADB 5.11.0_arenadata3** и выше.

В представлении *gp_toolkit.gp_bloat_expected_pages* некорректно сообщается о чрезмерности root-партиции таблицы, даже при отсутствии данных. В некоторых случаях представление *gp_toolkit.gp_bloat_diag* может возвращать ошибку integer вне диапазона (при обновлении с версии *5.3* и ниже).

Решение: Обновить представления в схеме *gp_toolkit* в новой базе данных.

- **29485 – Catalog and Metadata**

Когда сессия создает временные объекты в базе данных, **ADB** может не удалять временные объекты при завершении сессии в случае, если она прерывается или прекращается по команде администратора.

- **29496 – gpconfig**

Для небольшого числа параметров конфигурации сервера, таких как *log_min_messages*, команда *gpconfig -s <config_param>* не отображает правильное значение параметра для хостов сегмента, когда значение параметра на master отличается от значения на сегментах.

Для параметров с набором классификации master утилита отображает значение, заданное на мастере, как для мастера, так и для сегментов. Для этих параметров значение на мастере передается как часть запросов на инстансы сегмента. SQL-запрос, который **gpconfig** запускает для отображения значений параметра мастера и сегмента, возвращает значение мастер-хоста, которое передается сегменту как часть запроса.

Для параметров, таких как *log_min_messages*, инстансы сегмента используют значение хоста сегмента, указанное в файле *postgresql.conf* при запуске. Значение сегмента может быть переопределено для области запроса.

Решение: Для отображения значения параметра, указанного в *postgresql.conf* на мастер-хосте и хостах сегмента, можно указать в **gpconfig** опцию *–file*.

- **29246 – gpconfig**

При запросе параметра конфигурации сервера *gp_enable_gpperfmon* с помощью **gpconfig -s gp_enable_gpperfmon** *gpconfig* всегда выдает отчет *off*, даже если параметр установлен в значение *on* и запущены процессы агента *gpmmon* и *gpsmon*.

Решение: чтобы узнать, включен ли **gpperfmon**, проверить значение параметра конфигурации *gp_enable_gpperfmon* в файле *postgresql.conf* или использовать команду *ps* для поиска процессов *gpmmon* (мастер) или *gpsmon* (сегмент).

- **29395 – DDL**

Утилита **gpdbrestore** и **gprestore** завершается сбоем при попытке восстановить таблицу из резервной копии, которая неправильно определена с повторяющимися столбцами в качестве ключей распределения. Проблема возникает в результате некорректного определения таблицы утилитой **gpcrondump** или **gpbackup** при копировании. Команда *CREATE TABLE AS* создает такую неправильно определенную таблицу с помощью политики распределения.

Решение: Проблема *CREATE TABLE ISSUE* устранена. Команда *CREATE TABLE AS* не создает указанный тип таблицы, а возвращает ошибку. Однако попытка восстановления неверно определенной таблицы из резервной копии продолжает завершаться сбоем.

- **29351 – gptransfer**

Утилита **gptransfer** может скопировать строку данных с максимальной длиной *256 МБ*.

- **158011506 – Catalog and Metadata**

В некоторых случаях часовой пояс, используемый в базе данных **ADB**, может отличаться от часового пояса хост-системы или заданного пользователем. В некоторых редких случаях время, используемое и отображаемое в базе данных **ADB**, может немного отличаться от времени хост-системы.

Часовой пояс, используемый в базе данных **ADB**, выбирается из набора встроенных часовых поясов PostgreSQL. База данных выбирает часовой пояс путем сопоставления часового пояса PostgreSQL с часовым поясом, заданным пользователем, или часовым поясом хост-системы. Часовой пояс по умолчанию использует алгоритм для выбора часовых поясов PostgreSQL на основе часового пояса хост-системы. Если системный часовой пояс содержит информацию о секунде координации, база данных не может сопоставить системный часовой пояс с часовым поясом PostgreSQL.

Решение: Установить параметры базы данных **ADB** и часовых поясов хост-системы в часовой пояс, который поддерживается как базой данных, так и хост-системой. Например, с помощью утилиты **gpconfig**. Следующие команды показывают часовой пояс базы данных и устанавливают часовой пояс для US/Pacific:

```
# gpconfig -s TimeZone
# gpconfig -c TimeZone -v 'US/Pacific'
```

После изменения часового пояса необходимо перезапустить **ADB** (команда *gpstop -ra*).

В каталоге *pg_timezone_names* предоставляется информация о часовом поясе базы данных **ADB**.

- **N/A – PXF**

PXF доступен только для поддерживаемых платформ **Red Hat** и **CentOS**. **PXF** недоступен для платформ **SuSE**.

- **151135629 – Команда COPY**

Когда указано свойство *ON SEGMENT*, команда *COPY* не поддерживает *SELECT* в условии *COPY TO*. Например, следующая команда не поддерживается:

```
COPY (SELECT * FROM testtbl) TO '/tmp/mytst<SEGID>' ON SEGMENT
```

- **29064 – Storage: DLL**

Тип данных *money* принимает значения вне диапазона как отрицательные, и сообщение об ошибке не отображается.

Решение: Использовать только значения в диапазоне для типа данных *money* (64-разрядный для **ADB 5.x**). Или использовать альтернативный тип данных, например *numeric* или *decimal*.

- 3290 - **JSON**

Функция *to_json()* не реализована как вызываемая функция. Попытка вызова функции приводит к ошибке. Например:

```
tutorial=# select to_json('Fred said "Hi."'::text);
ERROR: function to_json(text) does not exist
LINE 1: select to_json('Fred said "Hi."'::text);
^
HINT: No function matches the given name and argument types.
You might need to add explicit type casts.
```

Решение: **ADB** вызывает функцию *to_json()* при приведении к типу данных *json*, поэтому следует выполнять листинг. Например: *SELECT '{"foo":"bar"}'::json;* база данных обеспечивает функции *array_to_json()* и *row_to_json()*.

- **148119917 – Resource Groups**

Тестирование функции **Resource Groups** выявило, что при использовании ядра по умолчанию в системе **RHEL/CentOS** может возникнуть ошибка. Проблема возникает из-за проблем в реализации контрольных групп ядра и приводит к ошибке ядра, похожей на следующую:

```
[81375.325947] BUG: unable to handle kernel NULL pointer dereference at 0000000000000010
      [81375.325986] IP: [<ffffffff812f94b1>] rb_next+0x1/0x50 [81375.326014] PGD 0 [81375.326025]
      Oops: 0000 [#1] SMP [81375.326041] Modules linked in: veth ipt_MASQUERADE
      nf_nat_masquerade_ipv4 iptable_nat nf_conntrack_ipv4 nf_defrag_ipv4 nf_nat_ipv4 xt_addrtype
      iptable_filter xt_conntrack nf_nat nf_conntrack bridge stp llc intel_powerclamp coretemp
      intel_rapl dm_thin_pool dm_persistent_data dm_bio_prison dm_bufio kvm_intel kvm crc32_pclmul
      ghash_clmulni_intel aesni_intel lrw gf128mul glue_helper ablk_helper cryptd iTCO_wdt
      iTCO_vendor_support ses enclosure ipmi_ssif pcspkr lpc_ich sg sb_edac mfd_core edac_core
      mei_me ipmi_si mei wmi ipmi_msghandler shpchp acpi_power_meter acpi_pad ip_tables xfs
      libcrc32c sd_mod crc_t10dif crct10dif_generic mgag200 syscopyarea sysfillrect crct10dif_
↪pclmul
      sysimgblt crct10dif_common crc32c_intel drm_kms_helper ixgbe ttm mdio ahci igb libahci drm␣
↪ptp
      pps_core libata dca i2c_algo_bit [81375.326369]  i2c_core megaraid_sas dm_mirror
      dm_region_hash dm_log dm_mod [81375.326396] CPU: 17 PID: 0 Comm: swapper/17 Not tainted
      3.10.0-327.el7.x86_64 #1 [81375.326422] Hardware name: Cisco Systems Inc
      UCSC-C240-M4L/UCSC-C240-M4L, BIOS C240M4.2.0.8b.0.080620151546 08/06/2015 [81375.326459]␣
↪task:
      ffff88140ecec500 ti: ffff88140ed10000 task.ti: ffff88140ed10000 [81375.326485] RIP:
      0010:[<ffffffff812f94b1>]  [<ffffffff812f94b1>] rb_next+0x1/0x50 [81375.326514] RSP:
      0018:ffff88140ed13e10  EFLAGS: 00010046 [81375.326534] RAX: 0000000000000000 RBX:
      0000000000000000 RCX: 0000000000000000 [81375.326559] RDX: ffff88282f1d4800 RSI:
      ffff88280bc0f140 RDI: 0000000000000010 [81375.326584] RBP: ffff88140ed13e58 R08:
      0000000000000000 R09: 0000000000000001 [81375.326609] R10: 0000000000000000 R11:
      0000000000000001 R12: ffff88280b0e7000 [81375.326634] R13: 0000000000000000 R14:
```

```
     0000000000000000 R15: 0000000000b6f979 [81375.326659] FS:  0000000000000000(0000)
     GS:ffff88282f1c0000(0000) knlGS:0000000000000000 [81375.326688] CS:  0010 DS: 0000 ES: 0000
     CR0: 0000000080050033 [81375.326708] CR2: 0000000000000010 CR3: 000000000194a000 CR4:
     00000000001407e0 [81375.326733] DR0: 0000000000000000 DR1: 0000000000000000 DR2:
     0000000000000000 [81375.326758] DR3: 0000000000000000 DR6: 00000000ffff0ff0 DR7:
     0000000000000400 [81375.326783] Stack: [81375.326792]  ffff88140ed13e58 ffffffff810bf539
     ffff88282f1d4780 ffff88282f1d4780 [81375.326826]  ffff88140ececae8 ffff88282f1d4780
     0000000000000011 ffff88140ed10000 [81375.326861]  0000000000000000 ffff88140ed13eb8
     ffffffff8163a10a ffff88140ecec500 [81375.326895] Call Trace: [81375.326912]
     [<ffffffff810bf539>] ? pick_next_task_fair+0x129/0x1d0 [81375.326940]  [<ffffffff8163a10a>]
     __schedule+0x12a/0x900 [81375.326961]  [<ffffffff8163b9e9>] schedule_preempt_disabled+0x29/
↪0x70
     [81375.326987]  [<ffffffff810d6244>] cpu_startup_entry+0x184/0x290 [81375.327011]
     [<ffffffff810475fa>] start_secondary+0x1ba/0x230 [81375.327032] Code: e5 48 85 c0 75 07 eb␣
↪19 66
     90 48 89 d0 48 8b 50 10 48 85 d2 75 f4 48 8b 50 08 48 85 d2 75 eb 5d c3 31 c0 5d c3 0f 1f 44
     00 00 55 <48> 8b 17 48 89 e5 48 39 d7 74 3b 48 8b 47 08 48 85 c0 75 0e eb [81375.327157] RIP
     [<ffffffff812f94b1>] rb_next+0x1/0x50 [81375.327179]  RSP <ffff88140ed13e10> [81375.327192]␣
↪CR2:
     0000000000000010
```

Решение: Обновить ядро **Red Hat** или **CentOS** до последней версии.

- **149789783 – Resource Groups**

Значительное снижение производительности **ADB** наблюдается при включении управления рабочими нагрузками **Resource Groups** на системах **RedHat 6.x**, **CentOS 6.x** и **SuSE 11**. Эта проблема вызвана ошибкой контрольной группы ядра **Linux**. Ошибка была исправлена в **CentOS 7.x** и **Red Hat 7.x**.

Когда **Resource Groups** включены в системах с поврежденным ядром, при запуске транзакции или запроса может быть задержка в *1* секунду или более. Задержка вызвана ошибкой ядра **Linux**, где механизм синхронизации вызывается *synchronize_sched* и выдает ошибку, когда процесс присоединяется к группе. См. http://www.spinics.net/lists/cgroups/msg05708.html и https://lkml.org/lkml/2013/1/14/97 для получения дополнительной информации.

Эта проблема приводит к тому, что операции одиночного присоединения занимают больше времени, а также, что все параллельные вложения выполняются последовательно. Например, одно приложение процесса может занимать около *0,01* секунды. При одновременном подключении *100* процессов самое быстрое присоединение занимает *0,01* секунды, а самое медленное - около *1* секунды. **ADB** выполняет присоединения процессов при запуске транзакции или запросов. Таким образом, ухудшение производительности зависит от одновременно запущенных транзакций или запросов и не связано с параллельными работающими запросами. Также **ADB** имеет оптимизацию для обхода переписывания, когда **QE** повторно используется несколькими запросами в том же сеансе.

Эта ошибка не влияет на системы **CentOS 7.x** и **Red Hat 7.x**

Решение: Если используется **RedHat 6** и производительность с **Resource Groups** приемлема, следует обновить ядро до версии *2.6.32-696* или выше, чтобы воспользоваться преимуществами и доработками новой реализации **Resource Groups**.

SuSE 11 не имеет версии ядра, решающей эту проблему; **Resource Groups** по-прежнему считаются экспериментальной функцией на данной платформе.

- **163807792 – gpbackup/gprestore**

Когда в качестве разделителя в текстовом формате внешней таблицы был указан знак процента *%*, **gpbackup** его неправильно экранировала в команде *CREATE EXTERNAL TABLE*. Проблема решена. Знак *%* правильно экранируется.

- **150906510 – Backup и Restore**

Резервные копии базы данных *4.3.15.0* и более поздних версий содержат следующую строку в файлах резервных копий:

```
SET gp_strict_xml_parse = false;
```

Однако, **ADB 5.0.0** не имеет параметр с *gp_strict_xml_parse*. При восстановлении данных из резервного набора версии *4.3* в кластере *5.0.0* может появиться предупреждение:

```
[WARNING]:-gpdbrestore finished but ERRORS were found, please check the restore report file
for details
```

Кроме того, файл отчета может содержать ошибку:

```
ERROR: unrecognized configuration parameter "gp_strict_xml_parse"
```

Данные предупреждения и ошибки не влияют на процедуру восстановления, и могут быть проигнорированы.

8.1 Обновление gp_toolkit.gp_bloat_diag

В **ADB 5.3.0** или более ранней версии **5.x** в некоторых случаях при выполнении запроса в отношении *gp_toolkit.gp_bloat_diag* появлялась ошибка целого числа вне диапазона. Проблема была решена в **ADB 5.4.0**. При обновлении базы данных **ADB** функция *gp_toolkit.gp_bloat_diag* и визуализация должны обновляться в базах данных, созданных с помощью **ADB 5.3.0** или более ранней версии **5.x**. Проблема была решена в **ADB 5.4.0** и более поздних версиях.

Чтобы проверить, требуется ли обновление для функции *gp_toolkit.gp_bloat_diag* и визуализации в базе данных, необходимо запустить **psql** и выполнить команду *df*, чтобы отобразить информацию о функции *gp_toolkit.gp_bloat_diag*:

```
df gp_toolkit.gp_bloat_diag
```

Если тип данных для *btdexppages* является целым числом (“numeric”), требуется обновление. Если тип данных является цифровым (“integer”), обновление не требуется. В примере тип данных *btdexppages* является целым числом и требует обновления:

```
List of functions
-[ RECORD 1 ]-------+-------------------------------------------------------------------------------
↪-------------------
Schema              | gp_toolkit
Name                | gp_bloat_diag
Result data type    | record
Argument data types | btdrelpages integer, btdexppages integer, aotable boolean, OUT bltidx␣
↪integer, OUT bltdiag text
Type                | normal
```

Необходимо запустить скрипт, чтобы обновить функцию и визуализацию и исправить проблему в каждой базе данных, которая была создана с помощью **ADB 5.3.0** или более ранней версии **5.x**. И в качестве пользователя *gpadmin* выполнить следующие действия:

1. Скопировать скрипт в текстовый файл в базе данных **ADB**.
2. Запустить скрипт в каждой базе данных, требующей обновления.

В примере обновлена функция *gp_toolkit.gp_bloat_diag* и визуализация в базе данных *mytest* и предполагается, что скрипт находится в файле *update_bloat_diag.sql* в доменном каталоге *gpadmin*:

```
psql -f /home/gpadmin/update_bloat_diag.sql -d mytest
```

Запустить скрипт в период низкой активности. Запуск скрипта в период высокой активности не влияет на функциональность базы данных, но может повлиять на производительность.

Скрипт gp_toolkit.gp_bloat_diag для обновления функций и представления

```
BEGIN;
CREATE OR REPLACE FUNCTION gp_toolkit.gp_bloat_diag(btdrelpages int, btdexppages numeric, aotable␣
↪bool,
    OUT bltidx int, OUT bltdiag text)
AS
$$
    SELECT
        bloatidx,
        CASE
            WHEN bloatidx = 0
                THEN 'no bloat detected'::text
            WHEN bloatidx = 1
                THEN 'moderate amount of bloat suspected'::text
            WHEN bloatidx = 2
                THEN 'significant amount of bloat suspected'::text
            WHEN bloatidx = -1
                THEN 'diagnosis inconclusive or no bloat suspected'::text
        END AS bloatdiag
    FROM
    (
        SELECT
            CASE
                WHEN $3 = 't' THEN 0
                WHEN $1 < 10 AND $2 = 0 THEN -1
                WHEN $2 = 0 THEN 2
                WHEN $1 < $2 THEN 0
                WHEN ($1/$2)::numeric > 10 THEN 2
                WHEN ($1/$2)::numeric > 3 THEN 1
                ELSE -1
            END AS bloatidx
    ) AS bloatmapping

$$
LANGUAGE SQL READS SQL DATA;

GRANT EXECUTE ON FUNCTION gp_toolkit.gp_bloat_diag(int, numeric, bool, OUT int, OUT text) TO␣
↪public;

CREATE OR REPLACE VIEW gp_toolkit.gp_bloat_diag
AS
    SELECT
        btdrelid AS bdirelid,
        fnnspname AS bdinspname,
        fnrelname AS bdirelname,
        btdrelpages AS bdirelpages,
        btdexppages AS bdiexppages,
        bltdiag(bd) AS bdidiag
    FROM
    (
        SELECT
            fn.*, beg.*,
            gp_toolkit.gp_bloat_diag(btdrelpages::int, btdexppages::numeric, iao.iaotype::bool)␣
↪AS bd
        FROM
            gp_toolkit.gp_bloat_expected_pages beg,
            pg_catalog.pg_class pgc,
            gp_toolkit.__gp_fullname fn,
```

```
            gp_toolkit.__gp_is_append_only iao

        WHERE beg.btdrelid = pgc.oid
            AND pgc.oid = fn.fnoid
            AND iao.iaooid = pgc.oid
    ) as bloatsummary
    WHERE bltidx(bd) > 0;

GRANT SELECT ON TABLE gp_toolkit.gp_bloat_diag TO public;
COMMIT;
```