Arenadata™ Database

Версия - v5.9.0-arenadata2

Примечания к выпуску ADB-5.9.0-arenadata2

2019 Arenadata™

Оглавление

1 Новые возможности **3**
- 1.1 Доработанная функция резервного копирования и восстановления объектов базы данных 3
- 1.2 Настройка PL/Container на ограниченный запуск контейнеров 4
- 1.3 Резерв ядер процессора для Resource Group 4
- 1.4 Предварительный запрос для PXF JDBC 4
- 1.5 Пул потоков для PXF JDBC 4

2 Измененные функции **6**

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

4 Поддерживаемые платформы **8**
- 4.1 Поддерживаемые платформы 9

5 Совместимость дистрибутивов Hadoop **10**
- 5.1 Совместимость дистрибутива Hadoop с помощью PXF 10
- 5.2 Совместимость дистрибутива Hadoop с помощью gphdfs 10

6 Обновление до ADB 5.9.0_arenadata2 **11**
- 6.1 Подготовка к обновлению 11
- 6.2 Обновление ADB с 5.x до 5.9.0_arenadata2 12
- 6.3 Устранение неполадок при неудачном обновлении 13

7 Решенные проблемы **14**

8 Известные проблемы и ограничения **17**
- 8.1 Обновление gp_toolkit.gp_bloat_diag 20

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 5.9.0_arenadata2. Для предыдущих версий продукта необходимо перейти к соответствующей документации

ADB 5.9.0_arenadata2 – выпуск с добавлением и изменением некоторых функций.

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

Глава 1

Новые возможности

ADB 5.9.0_arenadata2 включает в себя следующие новые функции:

- *Доработанная функция резервного копирования и восстановления объектов базы данных*
- *Настройка PL/Container на ограниченный запуск контейнеров*
- *Резерв ядер процессора для Resource Group*
- *Предварительный запрос для PXF JDBC*
- *Пул потоков для PXF JDBC*

1.1 Доработанная функция резервного копирования и восстановления объектов базы данных

База данных **ADB 5.9.0_arenadata2** включает доработки утилит резервного копирования **gpbackup** и восстановления **gprestore**.

Утилита **gprestore** поддерживает новые параметры:

- *–metadata-only* – восстанавливает таблицы из набора резервных копий, при этом данные таблицы не восстанавливаются;
- *–data-only* – восстанавливает данные таблицы из набора резервных копий без создания таблицы (таблица должна существовать в базе данных);
- *–jobs* – указывает количество заданий для параллельного выполнения при резервном копировании таблиц. При выполнении нескольких заданий каждое создает резервные копии таблиц в отдельной транзакции. Увеличение этого значения может повысить скорость резервного копирования данных.

Important: Утилита поочередно получает блокировку на каждую таблицу, подвергнутую резервному копированию, поэтому с ними не должно производиться никаких операций в процессе работы *gprestore*. При выполнении операций с таблицами ADB, находящимися в процессе резервного копирования, согласованность между их резервными копиями, полученными в рамках отдельных транзакций, не гарантирована

Для утилит **gpbackup** и **gprestore** параметры *–include-table* и *–include-table-file* поддерживают резервное копирование представлений и последовательностей. Теперь можно указать точное имя представления или последовательности.

Important: При использовании параметра *–include-table* или *–include-table-file* зависимые объекты автоматически не резервируются и не восстанавливаются, для этого необходимо явно указывать требуемые зависимые объекты

Например, при резервном копировании или восстановлении представления следует также указать таблицы, используемые им. А при резервном копировании или восстановлении таблицы, использующей последовательность, необходимо также явно указать эту последовательность.

1.2 Настройка PL/Container на ограниченный запуск контейнеров

Файл конфигурации для расширения языка **PL/Container** поддерживает новый атрибут *roles* для элемента XML *settings*. Данный атрибут управляет ролями (пользователями) базы данных **ADB**, которые могут запускать инстанс контейнера, управляемого конфигурацией среды выполнения. Значение атрибута *roles* – это имя роли (пользователя) базы данных **ADB** или список ролей, разделенных запятыми. В следующем примере элемента *settings* перечислены две *роли*:

```
<setting roles="gpadmin,user1"/>
```

PL/Container запускает контейнер, который использует конфигурацию среды выполнения только для перечисленных ролей. Если атрибут не указан, любая роль (пользователь) базы данных **ADB** может запускать инстанс конфигурации среды выполнения контейнера. Например, создается UDF с установкой языка *plcontainer* и определением среды выполнения *# container: runtime* с атрибутом *roles*. Когда роль (пользователь) базы данных выполняет UDF, **PL/Container** проверяет список ролей и запускает контейнер только в том случае, если выполняющая UDF роль находится в данном списке.

1.3 Резерв ядер процессора для Resource Group

С помощью базы данных **ADB** версии *5.9.0_arenadata2* можно зарезервировать определенные ядра центрального процессора для Resource Group. Выбор данного режима управления ресурсами ЦП позволяет гарантировать, что ресурсы процессора всегда доступны выбранной группе. Например, можно выделить фиксированное ядро ЦП Resource Group, используемой исключительно для коротких запросов.

1.4 Предварительный запрос для PXF JDBC

Перед выполнением любого запроса плагин **PXF JDBC** может выполнить произвольный SQL-код. Для этого следует передать его как обычную строку (включая символ “*;*” при необходимости) в качестве параметра *PRE_SQL* при создании внешней таблицы **ADB**.

Если исполнение “основного” запроса *INSERT* или *SELECT* не требуется, то в случае ошибки при выполнении произвольного кода в опции *PRE_SQL* можно установить параметр *STOP_IF_PRE_FAILS* равным любому значению. Тогда плагин **PXF JDBC** не исполняет “основной” запрос в случаях, когда произвольный код приводит к возникновению исключения во внешнем источнике данных. Если данный параметр не задан, плагин **PXF JDBC** продолжает выполнение “основного” запроса, даже если предварительный запрос завершается неудачно, и в журналы **PXF** помещается предупреждение.

1.5 Пул потоков для PXF JDBC

Запросы *INSERT* могут обрабатываться несколькими потоками, что значительно увеличивает производительность в случае, если внешняя база данных может эффективно работать с несколькими соединениями одновременно.

Вместе с пулом потоков рекомендуется использовать пакетирование. Тогда каждый поток получает данные из одной (целой) партии и обрабатывает ее. А если пул потоков используется без пакетной обработки, то каждый поток в пуле получает ровно одну запись; как правило, это занимает гораздо больше времени, чем обычный однопоточный *INSERT*.

Important: Если при выполнении операции любой поток из пула не смог выполнить какой-либо запрос, пользователь получает сообщение об ошибке. Тем не менее, как и в случае использования пакетной обработки, при возникновении ошибки часть записей может оказаться во внешней базе данных. Иными словами, операция вставки при использовании пула потоков не является атомарной даже в том случае, когда внешняя база данных поддерживает транзакции

Для включения пула потоков необходимо создать внешнюю таблицу с параметром *POOL_SIZE* с одним из следующих значений:

- *integer* > 1 – пул потоков состоит из заданного количества потоков;
- *integer* < 1 – количество потоков в пуле равно количеству процессоров в системе;
- *integer* $= 1$ – не использовать пул потоков.

Если параметр не указан, пул потоков не используется.

Глава 2

Измененные функции

В **ADB 5.9.0_arenadata2** изменены следующие функции:

- Для повышения производительности утилита **gpbackup** не помещает суммарный размер базы данных в файл отчета при выполнении операции резервного копирования, которая включает или исключает таблицы и/или схемы (отфильтрованная резервная копия). В других случаях информация добавляется в файл отчета;
- С целью поддержки возможности восстановления внешних таблиц базы данных **ADB 4.3.x**, утилита **gprestore** устанавливает параметр конфигурации сервера *gp_ignore_error_table* в положение *on* во время операции восстановления. Параметр позволяет использовать внешние таблицы версии *4.3.x*, определенные с условием *LOG INTO error_table*, которые необходимо восстановить в базе данных версии *5.x*;
- Изменения в API-интерфейсе плагина хранилища;
- Включает версию **PgBouncer 1.8.1**, предоставляющую встроенную поддержку TLS и PAM и совместимую конфигурацию *pg_hba.conf* (в случае использования **PgBouncer** в предыдущих версиях **ADB** при обновлении базы данных необходимо также обновить и **PgBouncer**).

Глава 3

Ключевые отличия ADB от open-source аналога

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

- Скрипт упаковки и установки продукта;
- Поддержка сжатия **ZStandard**;
- Поддержка управления базой данных **ADB** с помощью **Zeppelin management tool**;
- Поддержка мониторинга и управления запросами с помощью **Advanced monitoring stack**;
- Поддержка двустороннего обращения к источникам и приёмникам данных через **JDBC**;
- Поддержка федеративных запросов;
- Поддержка передачи предикатов во внешние системы.

Глава 4

Поддерживаемые платформы

ADB 5.9.0_arenadata2 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 5.9.0_arenadata2 поддерживает следующие версии **Java**:

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

ПО **ADB 5.9.0_arenadata2**, 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 5.9.0_arenadata2 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**.

4.1 Поддерживаемые платформы

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 5

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*

5.1 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.

5.2 Совместимость дистрибутива Hadoop с помощью gphdfs

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

Таблица5.1.: 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 6

Обновление до ADB 5.9.0_arenadata2

В разделе описывается путь обновления от версии **ADB 5.x** до **ADB 5.9.0_arenadata2**:

- *Подготовка к обновлению*
- *Обновление ADB с 5.x до 5.9.0_arenadata2*
- *Устранение неполадок при неудачном обновлении*

6.1 Подготовка к обновлению

Перед началом процесса обновления следует выполнить следующие рекомендации:

- Проверить работоспособность хост-оборудования **ADB** и убедиться, что хосты соответствуют требованиям для работы базы данных. Утилита **gpcheckperf** может помочь в подтверждении требований к хосту. Рекомендуется запускать утилиту **gpcheckcat** за несколько недель до обновления и, в случае обнаружения проблем, будет возможность разрешить их до запланированного обновления.

Утилита находится в *$GPHOME/bin*. Рекомендуется, чтобы **ADB** находилась в ограниченном режиме при запуске утилиты **gpcheckcat**. Если **gpcheckcat** сообщает о несоответствиях каталогов, можно запустить ее с параметром *-g* для создания SQL-скриптов для устранения несоответствий.

После запуска SQL-скриптов необходимо снова запустить **gpcheckcat**. Возможно, придется повторить процесс запуска **gpcheckcat** и создания SQL-скриптов, чтобы убедиться в отсутствии несоответствий. Рекомендуется, чтобы SQL-скрипты, созданные **gpcheckcat**, запускались в статистической системе, так как в ином случае утилита может сообщать ложные предупреждения.

Important: Если утилита gpcheckcat сообщает об ошибках, но не создает SQL-скрипт для их исправления, необходимо обратиться в службу поддержки ADB – e-mail: info@arenadata.io

---

- Во время процесса миграции с **ADB 5.0.0** создается резервная копия некоторых файлов и каталогов в *$MASTER_DATA_DIRECTORY*. Рекомендуется, чтобы файлы и каталоги, которые не использовались в **ADB**, при неободимости были скопированы и затем удалены из файла *$MASTER_DATA_DIRECTORY* перед миграцией.

Если при предыдущей установке **ADB** была настроена платформа **PXF**, необходимо остановить **PXF** и создать резервные копии файлов ее конфигурации до перехода на новую версию **ADB**. В случае если использование **PXF** не планируется и платформа не была настроена ранее, никаких действий не требуется.

6.2 Обновление ADB с 5.x до 5.9.0_arenadata2

Обновление версии с *5.x* до *5.9.0_arenadata2* включает в себя остановку **ADB**, обновление бинарных файлов базы данных, обновление **ADB** и ее перезапуск.

При наличии баз данных, созданных с помощью **ADB 5.3.0** или более ранней версии *5.x*, необходимо обновить функцию *gp_bloat_diag* и визуализацию в схеме *gp_toolkit* (дополнительная информация приведена в разделе Обновление gp_toolkit.gp_bloat_diag).

Если в **ADB** установлена база данных *gpperfmon*, процесс миграции изменяет ключ распределения таблиц *log_alert_** на столбец *logtime*. Перераспределение данных таблицы может занять некоторое время при первом запуске **ADB** после миграции.

Порядок действий для обновления базы данных **ADB** с версии *5.x* до *5.9.0_arenadata2*:

1. Войти в мастер-узел **ADB** от администратора:

   ```
   $ su - gpadmin
   ```

2. Выполнить интеллектуальное завершение работы текущей системы **ADB 5.x** (чтобы не было активных подключений). В примере используется опция *-a* для отключения запросов на подтверждение:

   ```
   $ gpstop -a
   ```

3. Запустить бинарную программу установки для *5.9.0_arenadata2* на мастер-хосте **ADB**. Указать место установки в том же базовом каталоге, что и текущая инсталляция. Например:

   ```
   /usr/lib/gpdb
   ```

Если **ADB** устанавливается с *rpm* (с правами администратора *root*), каталог установки – */usr/lib/gpdb*. Для установки *rpm* необходимо обновить права для новой сборки. Например, запустить команду как *root*, чтобы сменить пользователя и группу устанавливаемых файлов на *gpadmin*:

```
# chown -R gpadmin:gpadmin /usr/lib/gpdb
```

4. Если для развертывания **ADB** используется аутентификация **LDAP**, следует вручную отредактировать файл */usr/lib/gpdb/greenplum_path.sh*, добавив строку:

   ```
   export LDAPCONF=/etc/openldap/ldap.conf
   ```

5. Отредактировать среду суперпользователя (*gpadmin*) **ADB** и убедиться, что для новой установки используется файл *greenplum_path.sh*. Проверить на корректность следующую строку в *.bashrc* или в выбранном файле профиля:

   ```
   source /usr/lib/gpdb/greenplum_path.sh
   ```

Если в файлах профиля используется символическая ссылка (*/usr/lib/gpdb*), необходимо ее обновить и указать новую установленную версию.

6. Среда только что отредактированного исходного файла:

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

7. Запустить утилиту **gpseginstall** для установки бинарных файлов *5.9.0_arenadata2* на всех узлах сегмента, указанных в *hostfile*. Например:

   ```
   $ gpseginstall -f hostfile
   ```

8. Если ранее использовались любые расширения **ADB**, такие как **pgcrypto**, **PL/R**, **PL/Java**, **PL/Perl** и **PostGIS**, следует загрузить соответствующие пакеты и установить их с помощью утилиты **gppkg**.

9. Если в предыдущей установке базы данных **ADB** был настроен **PgBouncer**, необходимо перейти на новый **PgBouncer**.

10. После обновления всех хостов сегмента войти в систему как пользователь *gpadmin* и перезапустить **ADB**:

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

10. Если в предыдущей версии **ADB** была настроена **PXF**, необходимо повторно инициализировать платформу после обновления базы данных.

6.3 Устранение неполадок при неудачном обновлении

При возникновении проблем во время процесса миграции и наличии прав на **ADB**, приобретенных у **Arenadata**, необходимо обратиться в техническую поддержку **Arenadata** по адресу info@arenadata.io и предоставить следующую информацию:

- Завершенный процесс обновления (*Обновление ADB с 5.x до 5.9.0_arenadata2*);
- Данные журнала из *gpcheckcat* (находится в *~/gpAdminLogs*).

Глава 7

Решенные проблемы

В разделе перечислены проблемы в **ADB 5.9.0_arenadata2**, их решения и результат.

- 4938 – Query Optimizer

GPORCA генерировала *SIGSEGV* при обработке условия *LIKE* и выводила дополнительные ограничения и предикаты. Структура вывода ограничений **GPORCA** неправильно обрабатывала условие *LIKE*.

Проблема решена. Структура вывода ограничений **GPORCA** улучшена для игнорирования неподдерживаемых функций, таких как условие *LIKE*.

- 4909 – pg_dump

Утилита **pg_dump** неправильно включала информацию протокола базы данных **ADB** при выполнении операции резервного копирования таблиц или схем (например, при запуске команды *pg_dump -s -t*).

Проблема решена. Утилита не содержит протоколов при резервном копировании таблиц или схем.

- 1621 – PXF

PXF мог возвращать ошибку нехватки памяти при обработке большой рабочей нагрузки **Hadoop** при включенном олицетворении пользователя **PXF**.

Проблема решена. **PXF** корректно очищает записи кэша файловой системы в конце каждого запроса, чтобы они не накапливались и не вызывали ошибки нехватки памяти.

- 29404 – External Tables

При выполнении операции *SELECT* во внешней веб-таблице, определенной с условием *LOG ERRORS*, в базе данных **ADB** создавался *PANIC*, так как определение указанного типа внешней веб-таблицы некорректно сохранялось в каталоге **ADB**.

Проблема решена. Новые внешние веб-таблицы, определенные с условием *LOG ERRORS*, корректно сохраняются в каталоге. Для существующих внешних веб-таблиц, определенных с условием *LOG ERRORS*, таблица должна быть удалена и повторно создана.

- 29392 – gpbackup/gprestore

Восстановление таблицы с помощью утилиты **gprestore** при наличии или отсутствии последовательностей осуществлялось только с помощью операции *–include-table-file*.

Проблема решена. Утилиты **gpbackup** и **gprestore** расширены для поддержки восстановления зависимых объектов опциями *–include-table-file* и *–include-table*.

- 29388 – DDL

Создание новой базы данных завершалось неудачно, когда команда *CREATE DATABASE* использовалась с условием *TEMPLATE* и с содержанием более, чем *1 ГБ* данных в сегменте шаблона БД. Операция *CREATE DATABASE* неправильно копировала расширенные связи кучи из базы данных шаблона.

Проблема решена. Расширенные связи кучи правильно копируются из шаблона базы данных.

- 29387 – Query Optimizer

GPORCA создавала *PANIC* для некоторых запросов, содержащих условие *WITH ... UNION ALL* и вычисляемый столбец, используемый в условии *WITH*. В ходе процесса оптимизации запросов **GPORCA** не выполняла правильную настройку предикатов.

Проблема решена. **GPORCA** улучшена для правильного выполнения предиката для заданного типа запроса.

- 29385 – Query Optimizer

Для некоторых запросов, содержащих сравнение отрицания (сравнение с префиксом оператора *NOT*), ссылаемое на несколько столбцов, **GPORCA** генерировал *PANIC*. Например, сравнение отрицаний *NOT(a.f2=0 OR a.f3=0)* ссылается на два столбца: *f2* и *f3*.

Проблема решена. **GPORCA** правильно оценивает сравнения.

- 29373 – gpbackup/gprestore

Утилита **gprestore** не удавалась при попытке восстановления роли базы данных **ADB**, когда имя роли начиналось с цифры. Утилита **gpbackup** неправильно обрабатывала такую роль во время операции резервного копирования.

Проблема решена. Утилиты корректно обрабатывают указанный тип ролей.

- 29352 – gpbackup/gprestore

Выполнение операции **gprestore** не удавалось, если в резервной копии содержались значения параметров базы данных, длина которых превышала *64* символа. Например, параметр конфигурации сервера базы данных **ADB** *gp_default_storage_options* может иметь значение длиной более 64 символов.

Проблема решена. Улучшена обработка строк длиной более *64* символов с помощью **gpbackup** и **gprestore**.

- 29334 – gptransfer

Утилита **gptransfer** базы данных **ADB** не копировала последовательность в случае, если она была указана как значение по умолчанию. Утилита копировала только последовательность, которая создавалась при определении таблицы со столбцом с типом данных *serial*.

Проблема решена. Утилита передает указанные последовательности для таблиц.

- 29330 – gphdfs

В некоторых случаях внешняя таблица, определенная с помощью протокола **gphdfs**, генерировала ошибку при чтении файла *Parquet*. Ошибка возникала при обработке сложных типов с необязательными полями, а *Parquet*-данные не содержали данных для необязательных полей.

Проблема решена. Протокол **gphdfs** расширен для обработки указанного типа данных *Parquet*.

- 29329 – Query Planner

Для некоторых запросов к нескольким таблицам, содержащим предикат с сравнением *IN* или *ANY*, планировщик базы данных **ADB** генерировал *PANIC*. Реакция была вызвана потому, что планировщик неправильно оформлял вложенные запросы, созданные процессом оптимизации запросов.

Проблема решена. Теперь Query Planner правильно обрабатывает сгенерированные подзапросы и генерирует действительный план.

- 29308 – Catalog and Metadata

В некоторых ситуациях база данных **ADB** не позволяла правильно добавлять партиции в партицированные таблицы и возвращала очень много ошибок.

Проблема решена. Управление партицированными таблицами улучшено при добавлении партиций.

- 29307 – Query Planner

Для некоторых запросов к партицированной таблице база данных **ADB** генерировала *PANIC* при создании плана запроса. Планировщик запросов неправильно обращался к статистике столбцов из конечной партицированной таблицы, если столбец был удален из партицированной таблицы, а затем партиция была разделена.

Проблема решена. Планировщик запросов улучшен для правильного доступа к статистике столбцов в указанной ситуации.

- 29288, 29312 – gptransfer

Утилита **gptransfer** неправильно копировала последовательности таблиц (последовательности, созданные при определении столбца таблицы с типом данных *SERIAL*). Утилита сбрасывала поле последовательности *last_value* в значение *1*.

Проблема решена. Утилита правильно копирует последовательности таблиц.

- 29284 – PL/Python

Запросы в системе **ADB**, выполняемые функцию *PL/Python*, не могли быть отменены с помощью *pg_cancel_backend* или *pg_terminate_backend*. Функция *PL/Python* не распознавала сигнал отмены от *pg_cancel_backend*.

Проблема решена. *PL/Python* улучшена для распознавания сигнала отмены от *pg_cancel_backend*.

- 29262 – Interconnect

Когда система базы данных **ADB** была настроена на использование протокола **TCP** (*gp_interconnect_type – TCP*), некоторые запросы не были завершены, и в базе данных регистрировались ошибки соединения, в которых указано, что соединение было закрыто досрочно. Проблема была вызвана тем, что **ADB** неправильно управляла закрытием TCP-подключения.

Проблема решена. В базе данных **ADB** улучшен способ закрытия TCP-соединений.

- 157550115 – COPY

Команда *COPY ... ON SEGMENT* сообщала о некорректном количестве строк при копировании. Проблема возникала из-за того, что количество строк использовало 32-разрядное целое число.

Проблема решена. Счетчик строк расширен, чтобы использовать 64-битное целое число.

- 156027741 – Query Optimizer

GPORCA генерировала *PANIC* при доступе к статистике для столбцов, когда типы данных статистики не соответствовали типу столбца, определяемому DDL.

Теперь в указанной ситуации **GPORCA** создает набор статистических данных по умолчанию и выдает предупреждение для выполнения операции *ANALYZE* для исправления проблемы.

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**:

- **29347 - PgBouncer**

Когда база данных **ADB** настроена с использованием менеджера соединений **PgBouncer**, **PgBouncer** иногда возвращает ошибку *invalid server parameter*. Ошибка возникает, когда **PgBouncer** неправильно обрабатывает команды *SET*.

- **29351 - gptransfer**

Утилита **gptransfer** может копировать строки данных с максимальной длиной *256 МБ*.

- **158011506 - Catalog и 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 системы хоста.

Часовой пояс, используемый в базе данных **ADB**, выбирается из набора встроенных часовых поясов **PostgreSQL**. Кластер выбирает часовой пояс, сопоставляя часовой пояс **PostgreSQL** с часовым поясом хост-системы или указанным пользователем. При выборе часового пояса по умолчанию в базе данных **ADB** используется алгоритм выбора часовых поясов **PostgreSQL** на основе таковых в хост-системе. Если в хост-системе в часовом поясе включена информация leap second, кластер **ADB** не может соответствовать системному часовому поясу. Кластер вычисляет наилучшее соответствие с часовым поясом **PostgreSQL** на основе информации из хост-системы.

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

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

После изменения часового пояса необходимо перезапустить кластер при помощи команды *gpstop -ra*. В каталоге *pg_timezone_names* представлена информация о часовом поясе базы данных **ADB**.

- **26589 - Storage: Catalog и Metadata**

База данных **ADB** не блокирует схемы при создании таблицы внутри схемы. Одновременные операции *DROP* и *CREATE TABLE* приводят к утечке данных (к несвязанной таблице или потере связей между столбцами) в системном каталоге и, возможно, на диске. Например, отношение, созданное как *CREATE TABLE foobar (col1 int)*, при одновременном *DROP* может привести либо к утечке данных в самой таблице, либо в объекте атрибута *col1*.

Это не влияет на будущие запросы. Несоответствия каталога можно обнаружить с помощью утилиты *gpcheckcat*.

Временное решение: чтобы заблокировать параллельную операцию *DROP*, необходимо нажать и удерживать блокировку *ROW SHARE* в строке таблицы каталога *pg_namespace*, которая соответствует схеме. Например, транзакция получает блокировку *ROW SHARE* в строке таблицы каталога каталога *pg_namespace* для схемы *my_schema*:

```
begin;
SELECT nspname FROM pg_namespace WHERE nspname = 'my_schema' FOR SHARE;
  ...

end;
```

Во время транзакции блокировка предотвращает падение схемы.

- **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.

- **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;
```