Arenadata™ Database

Версия - v5.4

Примечания к выпуску ADB-5.4.1

2018 Arenadata™

Оглавление

1 Экспериментальные функции **3**

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

3 Поддерживаемые платформы **5**
3.1 Поддерживаемые платформы 6

4 Совместимость дистрибутивов Hadoop **7**
4.1 Совместимость дистрибутивов Hadoop с помощью PXF 7
4.2 Совместимость дистрибутивов Hadoop с помощью gphdfs 7

5 Обновление до ADB 5.4.1 **8**
5.1 Подготовка 8
5.2 Обновление ADB с 5.x до 5.4.1 8
5.3 Устранение неполадок при неудачном обновлении 10

6 Обновление PXF **11**
6.1 Действия перед обновлением PXF 11
6.2 Обновление PXF 12

7 Решенные проблемы **13**
7.1 Обновление gp_toolkit.gp_bloat_diag 15

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 5.4.1. Для предыдущих версий продукта необходимо перейти к соответствующей документации

ADB 5.4.1 – это патч с исправлениями для восстанавления поддержки платформы PXF, которая отсутствовала в версии 5.4.0.

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

Глава 1

Экспериментальные 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 поможет стимулировать разработку этих функций, и они могут поддерживаться в будущих версиях продукта.

Important: Экспериментальные функции не рекомендуются и не поддерживаются для развертывания в продуктивной среде. Эти функции могут быть изменены или удалены из будущих версий продукта на основе дальнейшего тестирования и обратной связи. Более того, любые функции, которые могут быть видны в open source code, но при этом не описаны в документации к продукту, должны рассматриваться как экспериментальные и неподдерживаемые для использования в продуктивной среде

Основные экспериментальные возможности в **ADB 5.4.1** включают:

- *gpbackup* и *gprestore* – экспериментальные утилиты, предназначенные для повышения производительности, функциональности и надежности резервных копий по сравнению с *gpcrondump* и *gpdbrestore*.

  gpbackup использует блокировки *ACCESS SHARE* на отдельном уровне таблицы, а не *EXCLUSIVE* блокировки в таблице каталога *pg_class*. Это позволяет выполнять команды *DDL* во время резервного копирования (такие операции как *CREATE*, *ALTER*, *DROP* и *TRUNCATE*), если эти операции не нацелены на текущий набор резервных копий.

  В **ADB 5.4.0** утилиты *gpbackup* и *gprestore* были обновлены до версии *1.0.0-alpha.3* и используют один файл для глобальных и базовых метаданных. Эта версия несовместима с наборами файлов резервных копий, созданных с версиями *v1.0.0-alpha.1* или *v1.0.0-alpha.2*.

  Резервные файлы, созданные с помощью *gpbackup*, предназначены для обеспечения будущих возможностей восстановления отдельных объектов базы данных вместе с их зависимостями, такими как функции и требуемые пользовательские типы данных
- **Recursive WITH Queries** (общие выражения таблицы)
- **Resource groups** остаются экспериментальной функцией только на платформе **SuSE 11** из-за ограниченной функциональности групп в ядре. **SuSE 12** разрешает проблемы с группой **Linux**, которые вызывают ухудшение производительности при включенных в **ADB Resource groups**
- Запись текста и данных двоичного формата *SequenceFile* в **HDFS** с использованием **PXF**

Глава 2

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 3

Поддерживаемые платформы

ADB 5.4.1 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

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 PL/Container
- Платформу PXF

ADB 5.4.1 поддерживает следующие версии **Java**:

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

ПО **ADB 5.4.1**, 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 5.4.1 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**.

3.1 Поддерживаемые платформы

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

- Единственной файловой системой, поддерживаемой для работы с базой данных **ADB**, является 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 4

Совместимость дистрибутивов Hadoop

ADB обеспечивает доступ к **HDFS** с помощью *gphdfs* и платформы **PXF**.

4.1 Совместимость дистрибутивов 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.

4.2 Совместимость дистрибутивов Hadoop с помощью gphdfs

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

Таблица4.1.: 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. Для MapR 5.x при условии FORMAT команда CREATE EXTERNAL TABLE поддерживается только TEXT и CSV

Глава 5

Обновление до ADB 5.4.1

5.1 Подготовка

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 из **ADB 5.0.0** создается резервная копия некоторых файлов и каталогов в *$MASTER_DATA_DIRECTORY*. Рекомендуется, чтобы файлы и каталоги, которые не использовались в **ADB**, были скопированы и, если необходимо, удалены из файла *$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** (подробное описание приведено в разделе Обновление PXF). В случае если использование **PXF** не планируется и платформа не была настроена ранее, никаких действий не требуется.

5.2 Обновление ADB с 5.x до 5.4.1

Обновление версии с *5.x* до *5.4.1* включает в себя остановку **ADB**, обновление двоичных файлов базы данных, обновление и перезапуск **ADB**.

Important: Если имеются базы данных, созданные с помощью ADB 5.3.0 или более ранней версии релиза 5.x, необходимо обновить функцию gp_bloat_diag и схему gp_toolkit (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.

Для обновления **ADB** с версии *5.x* до *5.4.1* необходимо выполнить следующие шаги:

1. Войти в мастер-узел **ADB** как администратор:

   ```
   $ 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 двоичную программу установки для *5.4.1* на главном хосте **ADB**. При появлении запроса выбрать место установки в том же базовом каталоге, что и текущая установка. 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 установленных 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 условие 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. Только что отредактированный исходный файл:

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

7. Запустить утилиту **gpseginstall** для установки двоичных файлов *5.4.1* на всех узлах сегмента, указанных в *hostfile*. Например:

   ```
   $ gpseginstall -f hostfile
   ```

8. Использовать утилиту *gppkg* для установки расширений **ADB**. Если ранее использовались любые расширения **ADB**, такие как *pgcrypto*, *PL/R*, *PL/Java*, *PL/Perl* и *PostGIS*, следует загрузить соответствующие пакеты и установить их;

9. После того как все хосты сегментов были обновлены, можно войти в систему как пользователь *gpadmin* и перезапустить систему **ADB**:

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

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

5.3 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 5.4.1*);
- Данные журнала из *gpcheckcat* (находится в *~/gpAdminLogs*).

Глава 6

Обновление PXF

При использовании **PXF** в текущей инсталляции базы данных **ADB** необходимо выполнить апгрейд сервиса **PXF** при обновлении базы данных **ADB** до новой версии.

Процедура обновления **PXF** описывает, как обновить **PXF** в установке базы данных **ADB**. Данная процедура использует *PXF.from* для ссылки на текущую версию **PXF** и *PXF.to* для ссылки на версию **PXF**, установленную при апгрейде **ADB** до новой версии.

6.1 Действия перед обновлением PXF

Необходимо выполнить следующую процедуру перед обновлением базы данных **ADB** до новой версии:

1. Войти на главный узел базы данных **ADB** и настроить среду. Например:

```
$ ssh gpadmin@<gpmaster>
gpadmin@gpmaster$ . /usr/local/greenplum-db/greenplum_path.sh
```

2. Создать текстовый файл с перечнем хостов узла базы данных **ADB** (по одному имени узла в строке). Убедиться, что в файле нет пустых строк и лишних пробелов. Например, файл с именем *seghostfile* может включать:

```
seghost1
seghost2
seghost3
```

3. Запустить команду *pxf stop*, чтобы остановить **PXF** на каждом сегменте хоста. Например:

```
$ gpadmin@gpmaster$ gpssh -e -v -f seghostfile "/usr/local/greenplum-db/pxf/bin/pxf stop"
```

4. Создать резервную копию файлов конфигурации *PXF.from*, расположенных в каталоге *$GPHOME/pxf/conf/*. Эти файлы должны быть одинаковыми на всех хостах сегмента, поэтому их следует скопировать с одного из хостов. Например:

```
gpadmin@gpmaster$ mkdir -p /save/pxf-from-conf
gpadmin@gpmaster$ scp gpadmin@seghost1:/usr/local/greenplum-db/pxf/conf/* /save/pxf-from-conf/
```

5. Следует оборатить внимание на расположение любых файлов *JAR*, которые могли быть добавлены в файл инсталляции *PXF.from*. Сохранить копию этих JAR-файлов на случай их возможного удаления или изменения при обновлении версии базы данных;

6. Выполнить обновление базы данных **ADB** до новой версии (Обновление до ADB 5.4.1), а затем продолжить апгрейд **PXF** (Обновление PXF).

6.2 Обновление PXF

После обновления базы данных **ADB** до новой версии необходимо выполнить следующую процедуру для апгрейда и настройки программного обеспечения *PXF.to*:

1. Войти на главный узел базы данных **ADB** и настроить среду. Например:

```
$ ssh gpadmin@<gpmaster>
gpadmin@gpmaster$ . /usr/local/greenplum-db/greenplum_path.sh
```

2. Если файл конфигурации *pxf-env.sh* обновлен в инсталляции *PXF.from*, необходимо повторно применить эти изменения к файлу и скопировать обновленный файл *pxf-env.sh* на все хосты сегмента. Например, файл *seghostfile*, содержащий список узлов хоста сегмента в кластере **ADB** (по одному имени узла в строке):

```
gpadmin@gpmaster$ vi /usr/local/greenplum-db/pxf/conf/pxf-env.sh
gpadmin@gpmaster$ gpscp -v -f seghostfile /usr/local/greenplum-db/pxf/conf/pxf-env.sh =:/usr/
↪local/greenplum-db/pxf/conf/pxf-env.sh
```

3. Инициализировать **PXF** на каждом сегменте хоста, как описано в Инициализация PXF;
4. Если обновлены файлы конфигурации *pxf-profiles.xml*, *pxf-log4j.properties*, *pxf-private.classpath* или *pxf-public.classpath* в инсталляции *PXF.from*, необходимо повторно применить эти изменения и скопировать обновленный файл (файлы) всем хостам сегмента. И выполнить действия данной главы для аналогичной команды *gpscp*;
5. Если добавлены дополнительные JAR-файлы в инсталляцию *PXF.from*, скопировать их в соответствующий каталог в инсталляции *PXF.to* на каждом сегменте хоста;
6. Запустить **PXF** на каждом хосте сегмента, как описано в разделе Запуск PXF.

Глава 7

Решенные проблемы

В разделе перечислены проблемы в **ADB 5.4.1** и результат их решения.

База данных **ADB 5.4.0** не поддерживала функции **PXF** из-за регрессий, обнаруженных после выпуска. Поддержка **PXF** возобновлена в **ADB 5.4.1**.

- **26589 - Каталог и метаданные**

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 - 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.

7.1 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;
```