

TI MS Tool

Установка программного обеспечения

Оглавление

1	О документе	4
1.1	Краткая информация о продукте.....	4
2	Инструкция по установке.....	5
2.1	Системные требования для использования Продукта	5
2.1.1	Требования к вычислительным и дисковым ресурсам	5
2.1.2	Общие требования по настройке наблюдаемого кластера СУБД Greenplum	6
2.1.3	Требования для работы съёмника Ресурсных групп	7
2.1.4	Требования для работы съёмника "Температура данных"	7
2.1.5	Требования для работы съёмника "Размер данных"	8
2.1.6	Настройка docker на Windows	8
2.2	Установка и быстрое начало работы.....	9
2.2.1	Выполнить установку и первичный запуск.....	9
2.2.2	Опубликовать в Greenplum процедуру "Размер данных"	10
2.2.3	Опубликовать файл лицензии	10
2.2.4	Запуск приложения в "холостом" режиме	10
2.2.5	Подготовка строки пароля для конфиг-файла	10
2.2.6	Настройка сервера приложений (подготовка конфиг-файла)	10
2.2.7	Открыть дашборд в WebUI.....	11
2.3	Запуск и остановка	11
2.3.1	Для ОС Linux.....	11
2.3.2	Для ОС Windows	11
2.4	Установка демо-данных.....	12
2.4.1	Демо-данные съёмника РГ	12
2.4.2	Демо-данные съёмника Температуры данных.....	13
3	Конфигурирование	15
3.1	Съёмники на основе движка "greenplum6"	15
3.1.1	Съёмник "Ресурсные группы"	16
3.1.2	Съёмник "Температура данных"	16
3.1.3	Съёмник "Размеры данных"	17
3.2	Съёмники на основе движка "http"	17
3.2.1	Съёмник "Состояние PXF"	19
3.2.2	Съёмник "Метрики PXF"	19
3.3	Уровень журналирования (логирования).....	20
3.4	Реквизиты подключения к БД Продукта.....	20
4	Эксплуатационные операции	22
4.1.1	Журнал работы сервера приложений (nodejs).....	22
4.1.2	Определение реквизитов подключения к наблюдаемому кластеру Greenplum ...	22
4.1.3	Обновление и проверка лицензии Продукта	22
4.1.4	Переустановка дашборда	23
4.1.5	Зачистка дискового пространства от логов / старых данных Продукта	23
4.1.6	Переустановка Продукта	24
4.1.7	Деинсталляция (удаление) продукта	25

4.1.8	Резервное копирование и восстановление	26
4.1.9	Изменение порта доступа к WebUI (Grafana)	26
4.1.10	Подготовка строки пароля для конфиг-файла	26
4.1.11	Получение версии сервера приложений (контейнера nodejs)	27
5	Нештатные ситуации	28
5.1	<i>Не запускается Приложение.....</i>	<i>28</i>
5.2	<i>Не появляются данные в таблицах с показателями работы Greenplum.....</i>	<i>28</i>
5.3	<i>Не появляются данные в таблицах витрин данных.....</i>	<i>31</i>
5.4	<i>Нет данных в графиках Grafana.....</i>	<i>31</i>

1 О документе

Документ содержит описание процесса установки TI MS Tool, включая требования к ресурсам и среде установки, а так же типовые проблемы, их причины и способы решения,

1.1 Краткая информация о продукте

Продукт TI MS Tool предназначен для мониторинга состояния и истории работы кластеров СУБД Greenplum (и производных версий СУБД при условии соблюдения совместимости).

Продукт содержит:

- контейнер сервера приложений
- контейнер СУБД PostgreSQL (Citus)
- контейнер Grafana

Сервер приложений выполняет работу съёмников данных с заданной регулярностью, и вызов обновления аналитических витрин на основе снятых данных.

Включает две группы съёмников:

- ресурсные группы - данные настройки ресурсных групп кластера, показатели утилизация ресурсов и обработки запросов к СУБД
- температура данных - данные на основе системных счётчиков о характеристиках обращений к таблицам и партициям, кол-ву записей (тьюплов) и времени последнего выполнения VACUUM и ANALYZE

Доступна настройка одного или нескольких кластеров СУБД Greenplum, с одним или несколькими типами SQL съёмников.

Пользовательский интерфейс в среде WebUI Grafana предоставляет дашборды на основе данных витрин, сформированных на данных съёмников. Включают дашборды с графиками, отражающими:

- характеристики утилизации ресурсными группами аппаратных ресурсов серверов кластера (CPU, RAM) на сегментах и на мастере, а также характеристики обработки запросов (кол-во выполняемых запросов, кол-во выполненных запросов, кол-во запросов в очереди ожидания начала выполнения) и т.п.;
- характеристики утилизации ресурсными группами аппаратных ресурсов серверов кластера (CPU, RAM) на сегментах с разными статистиками по часовым интервалам;
- сегменты с отклонением утилизации CPU от среднего по кластеру больше заданного порога (одного среднеквадратичного отклонения).

2 Инструкция по установке

2.1 Системные требования для использования Продукта

Продукт запускается как комплекс docker контейнеров. На сервере или рабочей станции для работы Продукта требуется установка следующих продуктов:

- docker версии не ниже 20.0.0
- docker-compose версии не ниже 1.27.4

Продукт совместим с СУБД Greenplum 6 (в частности с версией 6.26) и с СУБД Greenplum 5 (в частности 5.28).

Проверена совместимость продукта с запуском в следующих операционных системах на процессорах x86:

- Ubuntu 22.04 , 20.4
- РЕД ОС 7.3
- Windows 10

Проверена совместимость продукта с запуском в следующих операционных системах на процессорах M1 с некоторыми ограничениями (после редактирования файла лицензии или конфига сервер приложений необходимо рестартовать для применения изменений):

- MacOS (протестировано на Sonoma 14.4)

2.1.1 Требования к вычислительным и дисковым ресурсам

Минимальные требования к выделяемым аппаратным ресурсам для запуска Продукта:

- дисковое пространство: 75 GB, HDD или SSD (предпочтительно). Размер указан без учёта зеркалирования / дублирования дисков.
- CPU: 5 ядер, более высокая частота оказывает значительное влияние в отличие от добавления кол-ва ядер
- RAM: 3 GB

Минимальные требования рассчитаны исходя из следующих ориентировочных характеристик нагрузки на Продукт:

- Одновременна работа с WebUI Grafana до 3 пользователей с обновлением всех графиков дашборда не чаще каждые 30 секунд
- Мониторинг до 3х кластеров СУБД Greenplum с регулярностью снятия характеристик утилизации СУБД раз в минуту
- Хранение данных в течение до 12 месяцев (хранение истории за 6 мес. и запас по времени работы на плановую архивацию / удаление более старых данных в 5 мес.)
- Суммарное кол-во сегментов в кластерах до 300
- Суммарное кол-во ресурсных групп в кластерах до 15

- Суммарное кол-во атомарных таблиц (кол-во таблиц и кол-во партиций) 100 тыс.
 - Кол-во не партиционированных таблиц - 5000
 - Кол-во партиционированных таблиц 940
 - Кол-во партиций 94000
- Постоянное кол-во сегментов с перекосом (более 1 среднеквадратичного отклонения от среднего) до 20

Рекомендуемые требования для работы со среднестатистическими кластерами промышленных кластеров Greenplum:

- дисковое пространство: 100 GB, SSD или NVME (предпочтительно). Размер указан без учёта зеркалирования / дублирования дисков.
- CPU: 6 ядер, более высокая частота оказывает значительное влияние в отличие от добавления кол-ва ядер
- RAM: 8 GB

Рекомендуемые требования для работы со повышенной нагрузкой (кластера Greenplum с более чем 500 млн таблиц и партиций, повышенная частота сбора срезов, кол-во активных пользователей Grafana более 3 и т.п.):

- дисковое пространство: 150 GB, SSD или NVME (предпочтительно). Размер указан без учёта зеркалирования / дублирования дисков.
- CPU: 8 ядер, более высокая частота оказывает значительное влияние в отличие от добавления кол-ва ядер
- RAM: 12 GB

2.1.2 Общие требования по настройке наблюдаемого кластера СУБД Greenplum

Для снятия данных необходима учётная запись технологического пользователя, имеющему права на подключение к СУБД и выборку из таблиц системных справочников СУБД (например - к БД Postgres).

Технологической учётной записи должны быть доступны ресурсы для выполнения запроса к системным справочникам каждую минуту с возвратом строк по кол-ву ресурсных групп в СУБД, с выполнением 99.5% запросов в пределах 30 секунд (время обработки запроса напрямую влияет на корректность собираемых и анализируемых данных). Рекомендуется выделение учётной записи в отдельную администраторскую ресурсную группу.

Для работы съёмников необходимы доступы к конкретным системным справочникам СУБД Greenplum - они описаны в разделах требований съёмников.

Ниже пример SQL запроса к СУБД Greenplum для создания такого пользователя с необходимыми правами доступа для всех типов съёмников. Значение пароля "ti_mstool_pw" из примера необходимо заменить согласно политикам среды использования Продукта:

```
-- роль и права роли
```

```
create role ti_mstool_reader_role;
grant connect on database postgres to ti_mstool_reader_role;

-- учётная запись и выдача ей роли
create user ti_mstool_reader with password 'ti_mstool_pw';
grant ti_mstool_reader_role to ti_mstool_reader;
```

2.1.3 Требования для работы съёмника Ресурсных групп

Дисковое пространство под собираемые данные зависят от следующих метрик:

- SEGMENT_CNT: кол-во сегментов в кластере
- GROUPS_CNT: кол-во ресурсных групп в кластере
- CLUSTER_CNT: кол-во кластеров (если считать за разное время и по кластерам в отдельности то 1)
- SEGMENTS_SKEW_CNT: кол-во сегментов с перекосом (выше или ниже рабочего коридора)
- DAYS_CNT: кол-во дней наблюдения
- ITERATIONS_CNT: кол-во снятий срезов в течение суток (штатно каждую минуту, $60 * 24$)

2.1.3.1 Требования по настройке СУБД Greenplum

Подключаемый к Продукту кластер СУБД Greenplum должен быть сконфигурирован для работы с Ресурсными группами (РГ), вместо настройки по умолчанию с Ресурсными Очередями.

Выделение CPU для РГ должны быть настроены через параметр `cpu_rate_limit`, без использования выделения конкретных ядер CPU настройкой `cpuset`.

Необходимы права на `select` из системных справочников (права на `select` к системным справочникам выдаются по умолчанию), пример выделения прав:

```
grant select on gp_toolkit.gp_resgroup_config, gp_toolkit.gp_resgroup_status to
ti_mstool_reader_role;
```

2.1.4 Требования для работы съёмника "Температура данных"

Дисковое пространство под собираемые данные для Температуры зависит от следующих метрик:

- NON_PART_TABLES_CNT: суммарное кол-во не партиционированных таблиц
- PART_TABLES_CNT: суммарное кол-во партиционированных таблиц
- PARTITIONS_CNT: суммарное кол-во партиций всех уровней (включая все субпартиции и "other")
- DAYS_CNT: кол-во дней наблюдения
- ITERATIONS_CNT: кол-во снятий срезов в течение суток (штатно 1)

2.1.4.1 Требования по настройке СУБД Greenplum

Необходимо чтобы был включен сбор системных статистик для таблиц (включено по умолчанию). Регулируется настройкой `"track_counts"`, значение должно быть `"on"`. Изменение настройки доступно только суперпользователям.

Необходимы права на select из системных справочников (права на select к системным справочникам выдаются по умолчанию), пример выделения прав:

```
grant select on pg_stat_user_tables, pg_catalog.pg_partitions , pg_catalog.pg_partition to ti_mstool_reader_role;
```

2.1.5 Требования для работы съёмника "Размер данных"

Дисковое пространство под собираемые данные для "Размера данных" зависит от следующих метрик:

- NON_PART_TABLES_CNT: суммарное кол-во не партиционированных таблиц
- PART_TABLES_CNT: суммарное кол-во партиционированных таблиц
- PARTITIONS_CNT: суммарное кол-во партиций всех уровней (включая все субпартиции и "other")
- DAYS_CNT: кол-во дней наблюдения
- ITERATIONS_CNT: кол-во снятий срезов в течение суток (штатно 1)

2.1.5.1 Требования по настройке СУБД Greenplum

Подсчёт размера данных выполняется с использованием пользовательской процедуры. Процедуру необходимо зарегистрировать в наблюдаемой БД в СУБД Greenplum от имени пользователя, имеющего права на создание external web table и обращение к операционной системе для получения информации о размере файлов данных СУБД Greenplum на мастере и на сегментах.

Процедуры (основная и вспомогательные) размещается в каталоге дистрибутива Продукта, файл `./resources/gp6/07.3.extractor-dataSize-gp-sp.sql`. Функция совместима и с Greenplum 5 старших подверсий, и с Greenplum 6.

Необходимы права на выполнение созданных функций:

```
grant execute on function public.ti_size_n_skew_core(p_out_common text, p_included text[], p_excluded text[]) to ti_mstool_reader_role;
grant execute on function public.ti_size_n_skew_core_res(p_included text[], p_excluded text[]) to ti_mstool_reader_role;
grant execute on function public.ti_size_n_skew() to ti_mstool_reader_role;
grant execute on function public.ti_size_n_skew(p_included text) to ti_mstool_reader_role;
grant execute on function public.ti_size_n_skew(p_included text[]) to ti_mstool_reader_role;
grant execute on function public.ti_size_n_skew(p_included text[], p_excluded text[]) to ti_mstool_reader_role;
```

2.1.6 Настройка docker на Windows

Для работы продукта на ОС Windows требуется настройка docker и docker-compose с WSL2. Текущие версии Docker Desktop требуют версии Windows 10 или Windows 11.

В случае если при установке Docker Desktop выдаётся сообщение о не выполнении минимальных требований - необходимо проверить версию Windows. Возможно, потребуется установка более старого дистрибутива Docker Desktop - в случае невозможности выполнения требований для установки актуальной версии.

Так версия Docker Desktop 4.30.0 требует от Windows 10 версии 21H1 не ниже 19044, в том время как более ранняя версия 4.24.0 работает с более поздней версией 19043.

Кроме того, для работы Docker Desktop на Windows потребуется WSL2, обновление WSL выполняется штатно следующим вызовом:


```
wsl --update
```

! Штатное обновление WSL до WSL2 может занимать длительное время - до нескольких часов.

После завершения установки потребуется выполнить запуск Docker Desktop.

2.2 Установка и быстрое начало работы

Дистрибутив представляет из себя архив ti-mstool-VERSION.tar (с указанием значения версии вместо VERSION).

Для установки необходимо разархивировать файл в целевой каталог (как zip архив), и выполнить последующие шаги (детально описано в подразделах):

- разместить лицензию
- выполнить "холостой" запуск продукта - без корректного конфига
- подготовить строку с паролем учётной записи для подключения к СУБД Greenplum в зашифрованном виде, для указания в конфигурационном файле сервера приложений
- указать реквизиты подключения к наблюдаемой СУБД Greenplum и ID для этого кластера
- рестартовать приложение для актуализации конфига
- войти в WebUI

2.2.1 Выполнить установку и первичный запуск

2.2.1.1 На Linux

Распаковать архив дистрибутива в заданный каталог, например следующими командами находясь в каталоге с файлом дистрибутива (включает создание каталога в /opt) - для условной версии 0.0.0-0 :

```
export VERSION=ti-mstool-0.0.0-0
sudo mkdir /opt/${VERSION}
sudo tar -xvzf ./${VERSION}.tar -C /opt/${VERSION}
sudo chmod 777 -R /opt/${VERSION}
cd /opt/${VERSION}/ti-mstool
```

Для установки и запуска выполнить команду

```
sudo ./ti-mstool-linux.sh install
```

В рамках установки выполняется загрузка докер-имиджей, создание докер-сети и запуск работы всех компонентов в фоновом режиме.

Проверку последующих шагов можно отслеживать по логу работы контейнера nodejs.ms-tool , для этого в отдельном терминале необходимо открыть его вывод в режиме отслеживания новых строк:

```
sudo docker logs -f nodejs.ti-mstool
```

2.2.1.2 На Windows

Распаковать архив дистрибутива - ниже описан вариант распаковки в Windows 10 для условного дистрибутива версии 0.0.0-0.

Для распаковки со штатным 7-zip архиватором Windows 10+ файл необходимо переименовать, добавив в конец расширение ".zip"

В Explorer в контекстном меню выбрать "7-Zip" и в подменю "Распаковать в "ti-mstool-0.0.0-0.tar\". Появится файл .tar - для него повторить процедуру, указав распаковку в каталог "ti-mstool-0.0.0-0\". В том же каталоге появится каталог ti-mstool-0.0.0-0 с подкаталогом ti-mstool, содержащим файлы продукта - каталогом установки считается "ti-mstool-0.0.0-0/ti-mstool" .

2.2.2 Опубликовать в Greenplum процедуру "Размер данных"

Для работы съёмника "Размер данных" необходимо опубликовать пользовательскую функцию в наблюдаемые БД, см. 0

2.2.3 Опубликовать файл лицензии

Единовременно активной у инсталляции продукта может быть только одна версия лицензии. В случае отсутствия валидной лицензии функции по снятию и трансформации данных недоступны.

Размещение файла лицензии и проверка её корректности описаны в п. 4.1.3

2.2.4 Запуск приложения в "холостом" режиме

Контейнеры автоматически стартуют по окончанию установки, штатно этот шаг выполняется автоматически.

Для подготовки паролей для конфига в зашифрованном виде - необходимо чтобы Продукт (конкретнее - его докер-контейнеры) были стартованы.

Если контейнеры не были стартованы, выполнить команду запуска:

```
sudo ./ti-mstool-linux.sh start
```

Более детальное описание запуска и установки см. 2.3

2.2.5 Подготовка строки пароля для конфиг-файла

Пароли в конфиг-файле сервера приложений (для подключения к Greenplum в частности) хранятся в файле в зашифрованном виде. Процедура подготовки пароля см. п. 4.1.10 - пароль в зашифрованном виде необходим для подготовки конфига.

Если необходимо создать технологическую учётную запись в Greenplum - см. 2.1.2

2.2.6 Настройка сервера приложений (подготовка конфиг-файла)

Настройка сервера приложений Продукта осуществляется через конфигурационный файл ./configs/config.json

Конфиг файл в формате текстового JSON файла, включает описание объекта настроек с несколькими группами параметров, см. п. 3

! В случае если файлы лицензии и/или конфига не менялись в режиме редактирования, а переносились / пересоздавались - необходимо выполнить рестарт приложения (остановку и запуск).

2.2.7 Открыть дашборд в WebUI

Открыть браузер и в нём открыть страницу с адресом "http://{ИМЯ ИЛИ АДРЕС СЕРВЕРА}:3000", в случае если вход осуществляется на локально развёрнутое приложение (например на том же ноутбуке), вместо имени или адреса можно указать строку "localhost".

По умолчанию создаётся учётная запись grafana с таким же паролем, с правами администратора. При необходимости создания пользователей с ограниченными правами (только на чтение) - необходимо выполнить настройки (регистрацию пользователей, интеграцию с внешними системами для проверки аутентификации и авторизации) с использованием штатных возможностей продукта Grafana версии 10.

В случае выхода новой версии дашборда или необходимости его переустановки - его необходимо установить из файла, см. 4.1.4 .

2.3 Запуск и остановка

2.3.1 Для ОС Linux

Первичный запуск выполняется сразу после установки docker компонент следующим вызовом, Он выполняет регистрацию докер-имиджей из файлов дистрибутива, создаёт докер-сеть. Вызовы выполняются в каталоге Продукта.

```
sudo ./ti-mstool-linux.sh install
```

Запуск в дальнейшем можно так же выполнять этим вызовом:

```
sudo ./ti-mstool-linux.sh start
```

Проверить работу контейнеров и утилизация ими ресурсов следующей командой - она покажет обновляемое состояние по выполняемым контейнерам:

```
sudo docker stats
```

Для проверки лога работы основного контейнера - сервера приложений - выполнить вывод лога контейнера на отдельный экран терминала в режиме отслеживания новых строк:

```
sudo docker logs -f nodejs.ti-mstool
```

Остановка выполняется вызовом stop:

```
sudo ./ti-mstool-linux.sh stop
```

2.3.2 Для ОС Windows

До запуска продукта требуется настройка работы docker и docker-compose, см. 2.1.4

Все шаги установки / запуска / остановки / удаления аналогичны запуску под Linux с отличием в вызовах команд скриптом ti-mstool-windows.bat вместо ti-mstool-linux.sh и все вызовы без sudo.

То есть вместо

```
sudo ti-mstool-linux.sh
```

вызывать

```
ti-mstool-windows.bat
```

2.4 Установка демо-данных

2.4.1 Демо-данные съёмника РГ

В демонстрационных целях продукт может быть инициализирован с демо-данными, позволяющими продемонстрировать работу кластера на сэмплированных данных 2х кластеров.

Для демонстрационных целей не требуется настройка конфига и лицензия - достаточно распаковать дистрибутив и выполнить установку и первичный запуск.

Для выполнения установки и запуска продукта достаточно выполнить генерацию демо-данных. По умолчанию данные генерируются за период в один месяц Январь 2024 года, но период может быть настроен непосредственно в скрипте-генераторе.

Запуск генерации данных выполняется следующим вызовом:

```
sudo docker exec postgres.ti-mstool psql -U ti_owner -d ti_mstool_db -f /docker-entrypoint-initdb.d/demo-data/03.demo-data-resourceGroups.sql
```

Выполнение занимает некоторое время (минуты), при штатном выполнении в консоль должны последовательно быть выведены сообщения с подобным содержанием:

```
You are now connected to database "ti_mstool_db" as user "ti_owner".
-----
[2024-07-11 14:04:04.475] Call data emulation for resource group extractor,
ti_mstool_db.sp_emul_rg [ 2024-01-01 00:00:00.000, 2024-02-01 00:00:00.000 ], 8 segments
table "temp_dict_resource_groupes" does not exist, skipping
-----
[2024-07-11 14:05:40.606] --- done -----

[2024-07-11 14:05:40.684] call ti_mstool_db.sp_dm_rg_hourly_upsert( gp-demo-2, 2024-01-01
00:00:00, 2024-01-31 23:59:00 )
[2024-07-11 14:07:00.855] call ti_mstool_db.sp_dm_rg_minutely_upsert( gp-demo-2, 2024-01-01
00:00:00, 2024-01-31 23:59:00 )
[2024-07-11 14:16:43.373] call ti_mstool_db.sp_dm_rg_skew_segments_upsert( 2024-01-01
00:00:00, 2024-01-31 23:59:00, gp-demo-2 )
[2024-07-11 14:16:51.545] done

[2024-07-11 14:16:51.819] call ti_mstool_db.sp_dm_rg_hourly_upsert( gp-demo-1, 2024-01-01
00:00:00, 2024-01-31 23:59:00 )
[2024-07-11 14:17:17.274] call ti_mstool_db.sp_dm_rg_minutely_upsert( gp-demo-1, 2024-01-01
00:00:00, 2024-01-31 23:59:00 )
[2024-07-11 14:25:08.962] call ti_mstool_db.sp_dm_rg_skew_segments_upsert( 2024-01-01
00:00:00, 2024-01-31 23:59:00, gp-demo-1 )
[2024-07-11 14:25:13.677] done
```

Для проверки успешности - необходимо открыть дашбор за период генерируемых данных, графики должны быть заполнены, пример см. рис. 1

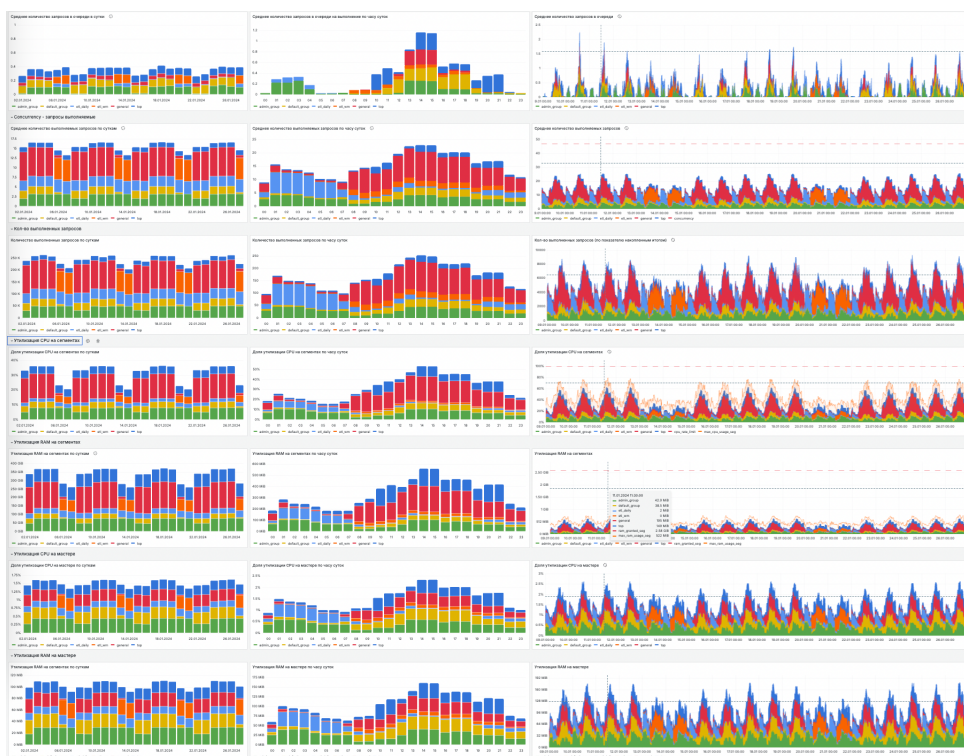


рис. 1 Пример дашборда со сгенерированными демо-данными

2.4.2 Демо-данные съёмника Температуры данных

Генерация демо-данных по съёмнику температуры данных формирует данные по демо-кластеру с ID "demo-temperature-01" за период с 2024-01-01 по 2024-06-01.

Для партиционированных записей считается что последняя партиция - текущий месяц, при заполнении истории в партициях, которые относительно отчётного момента находятся в будущем периоде, заполняются пустым значениями (нет активностей).

Эмулируются данные для четырёх таблиц:

Таблица	Описание	Операции ежедневно
schema02.table01_part	<p>Партиционированная, 12 месячных партиций.</p> <p>Профиль операций - только вставки и выборки, без изменения данных. Нормальный перекос.</p> <p>Активность в последние две партиции 100%, далее утихает до 10го месяца, после чего обнуляется.</p>	<p>Селекты от 1000 до 3000</p> <p>Вставки записей от 10000 до 25000</p> <p>Перекос от 0.05 до 0.1</p> <p>Без выполнения VACUUM и ANALYZE</p>
schema02.table02_part	<p>Партиционированная, 5 партиций.</p> <p>Профиль операций - смешанный, вставки, выборки, обновления и удаления. Большой перекос.</p> <p>Активность в последние 4 месяцев 100%, пятый 80%.</p>	<p>Селекты от 100 до 300</p> <p>Вставки записей от 300 до 650</p> <p>Обновления записей от 100 до 250</p> <p>Удаления записей от 300 до 650</p> <p>Перекос от 0.1 до 0.3</p> <p>Выполнение VACUUM каждый 3й день</p> <p>Выполнение ANALYZE каждый 5й день</p>

Таблица	Описание	Операции ежедневно
schema01.table01	Не партиционированная. Профиль операций - только вставки и выборки, без изменения данных. Нормальный перекос.	Селекты от 1000 до 3000 Вставки записей от 10000 до 25000 Перекос от 0.07 до 0.11 Без выполнения VACUUM и ANALYZE
schema01.table02	Не партиционированная. Профиль операций - смешанный, вставки, выборки, обновления и удаления. Большой перекос.	Селекты от 100 до 300 Вставки записей от 300 до 650 Обновления записей от 100 до 250 Удаления записей от 300 до 650 Перекос от 0.02 до 0.25 Выполнение VACUUM каждый 3й день Выполнение ANALYZE каждый 5й день

Вызов генерации выполняется следующей командой:

```
sudo docker exec postgres.ti-mstool psql -U ti_owner -d ti_mstool_db -f /docker-entrypoint-initdb.d/demo-data/06.12.demo-data-dataTemperature.sql
```

При штатном выполнении генерации демо-данных и витрин на их основе, в логе должны быть выведены сообщения подобные следующим:

```
[2024-07-05 18:34:49.712] Call ti_mstool_db.sp_emul_temperature [ 2024-01-01 00:00:00.000,
2024-06-01 00:00:00.000 ], cluster demo-temperature-01
table "demotempparts" does not exist, skipping
table "demotemptables" does not exist, skipping-----
[2024-07-05 18:34:52.476] Start ti_mstool_db.sp_dm_temp_upsert [ 2024-01-01 00:00:00, 2024-
06-01 00:00:00 ] (2023-12-31, 2024-05-31), cluster demo-temperature-01
table "t_data_dmttp" does not exist, skipping
[2024-07-05 18:34:52.490] Tables - non-partitioned
[2024-07-05 18:34:52.491] Partitions
[2024-07-05 18:34:52.500] Tables - partitioned
[2024-07-05 18:34:52.502] -- Done -----
```

3 Конфигурирование

В конфигурационном файле сервера приложений необходимо указать активные съёмники и подключение к СУБД Продукта.

Съёмники указываются сгруппированные согласно "движку" - "greenplum6" (на основе JDBC подключения к СУБД) либо "http" (на основе http / https запроса).

3.1 Съёмники на основе движка "greenplum6"

Получение данных реализовано за счёт запросов с использованием JDBC подключения к Greenplum 6 или Greenplum 5 (совместим с обеими версиями).

Параметры подключения к должны включать атрибуты, указанные в Таблица 1

Таблица 1 Параметры подключения съёмников движка greenplum 6

Атрибут	Описание	Пример значения
connection.host	IP адрес или полное доменное название сервера СУБД	"postgres.ti-mstool"
connection.port	порт подключения к СУБД	5432
connection.username	учётная запись пользователя СУБД	"ti_owner"
connection.password	пароль пользователя СУБД	"fe2520f302d96ccd64d29a8d907d3cc0351f06c2fef5aa54475bb6d743e8e4cf29d8f50d5c597ca6faa6"
connection.db	название БД	"ti_mstool_db"
extractors	Идентификаторы активных съёмников для данного кластера Greenplum - один или несколько. Может принимать следующие варианты значений: <ul style="list-style-type: none">строкой - если съёмник один и его настройки дефолтовыеJSON объектом - если съёмник один и его настройки переопределяются JSON массивом - строк или объектов Детально варианты пользовательских настроек описаны в подразделах ниже.	"resourcesGroup" ["resourcesGroup", "dataTemperature"] { "id": "dataTemperature", "cronUpdateRate": "1 * * * *" } ["resourcesGroup", { "id": "dataTemperature", "cronUpdateRate": "1 * * * *" }]
additionalData.cluster_id	Идентификатор кластера СУБД Greenplum, который будет отображаться в WebUI Продукта для наблюдаемого кластера. Обязательный - с этим идентификатором кластера данные далее будут доступны в БД и WebUI	"gp-01"
isDisabled	Опционально. Признак игнорирования регистрации в конфиге, false (по умолчанию) или (true).	true

Фрагмент конфига съёмников для одного подключения к СУБД Greenplum с настройками работы съёмников по умолчанию приведён в примере ниже:

```

{
  "greenplum6": {
    "gp6-reg": {
      "connection": {
        "port": 5432,
        "host": "10.20.30.40",
        "db": "dbname",
        "user": "username",
        "password": "7b59...795e"
      },
      "additionalData": {
        "cluster_id": "gp6-id"
      },
      "extractors": [ "resourcesGroup", "dataTemperature", "dataSize" ]
    }
  }
}

```

3.1.1 Съёмник "Ресурсы группы"

В настройках ID съёмника - resourcesGroup.

Реализует снятие срезов данных о настройках ресурсных групп, включая:

- перечень и настройки системных групп - лимиты, конкурентность
- характеристик утилизации ресурсов CPU и RAM на мастере и сегментах
- характеристики обработки запросов очереди - сколько ожидает начала обработки, сколько обрабатывается, сколько суммарно обработано (накопленным итогом)

Штатная регулярность - каждую календарную минуту.

Параметры пользовательской настройки съёмника resourcesGroup см. Таблица 2.

Таблица 2 пользовательские настройки съёмника resourcesGroup

Атрибут	Описание	Пример значения
cronUpdateRate	Расписание работы съёмника	"1 * * * *"

3.1.2 Съёмник "Температура данных"

В настройках ID съёмника - dataTemperature.

Реализует снятие снимать данные о системных счётчиках по обращениям (сканированиям) атомарных таблиц, кол-ву вставленных и удалённых строк и характеристики атомарных таблицы. Включает данные о перекосе распределения записей между сегментами.

Параметры пользовательской настройки съёмника dataTemperature см. Таблица 3.

Таблица 3 пользовательские настройки съёмника dataTemperature

Атрибут	Описание	Пример значения
cronUpdateRate	Расписание работы съёмника	"1 * * * *"
exclude	Названия схем, исключаемых из собираемой информации. По умолчанию исключений нет	["sch_nonpart_0", "sch_npart_0"]
include	Названия схем, по которым собирается информация. По умолчанию включены все	["sch_nonpart_0", "sch_npart_0"]

	пользовательские схемы (т.е. кроме служебных - включая системные справочники)	
--	---	--

При необходимости фиксирования промежуточных значений внутри суток (чаще чем раз в сутки) - можно выставить, например, ежечасовую регулярность снятия данных. При этом производная витрина температуры будет заполняться за отчётный период в сутки, значение на начало суток и на конец суток будут определяться как наиболее близкие по времени значения в пределах текущих / предыдущих суток отчётного периода. Это позволит видеть в витрине актуализируемые промежуточные значения температуры за текущие не завершённые сутки, и значения входящих атрибутов на начало суток.

3.1.3 Съёмник "Размеры данных"

В настройках ID съёмника - dataSize.

Реализует снятие срезов данных о значениях системных счётчиков операций с таблицами и партициями таблиц:

- кол-во сканирований
- кол-во операций записи и удаления "тьюплов" (версий строк)
- кол-во живых и "мёртвых" тьюплов
- дата-время последних выполнений VACUUM и ANALYZE (включая автоматический)
- перекос распределения записей (не удалённых регистраций, "тьюплов") по сегментам

Штатная регулярность - каждый календарный день, в 02:05 суток с учётом выставленной на сервере приложений временной зоне.

Параметры пользовательской настройки съёмника dataSize см. Таблица 3.

Таблица 4 пользовательские настройки съёмника dataSize

Атрибут	Описание	Пример значения
cronUpdateRate	Расписание работы съёмника	"1 * * * *"
exclude	Названия схем, исключаемых из собираемой информации. По умолчанию исключений нет	["sch_nonpart_0", "sch_npart_0"]
include	Названия схем, по которым собирается информация. По умолчанию включены все пользовательские схемы (т.е. кроме служебных - включая системные справочники)	["sch_nonpart_0", "sch_npart_0"]

3.2 Съёмники на основе движка "http"

Съёмники выполняют http/https запрос, выполняют разуор ответа - согласно формату JSON либо формату Prometheus экспортера.

Параметры подключения к источнику должны включать либо один источник, либо массив сточников. Каждый источник описывается объектом, который должен включать атрибуты, указанные в Таблица 5

Таблица 5 Параметры подключения съёмников движка http

Атрибут	Описание	Пример значения
connection	Настройки подключения к источнику - один объект или массив объектов. см Таблица 6	{ "host": "192.168.10.20", "port": 5432} или [{ "host": "192.168.10.21", "port": 5432}, { "host": "192.168.10.22", "port": 5432}]
extractors	Идентификаторы активных съёмников для данного кластера Greenplum - один или несколько. Может принимать следующие варианты значений: <ul style="list-style-type: none"> • строкой - если съёмник один и его настройки дефолтовые • JSON объектом - если съёмник один и его настройки переопределяются JSON массивом - строк или объектов Детально варианты пользовательских настроек описаны в подразделах ниже.	"pxfPrometheus" или { "id": "pxfPrometheus", "cronUpdateRate": "*/20 * * * *"}
additionalData.cluster_id	Идентификатор кластера СУБД Greenplum, который будет отображаться в WebUI Продукта для наблюдаемого кластера. Обязательный - с этим идентификатором кластера данные далее будут доступны в БД и WebUI	"gp-01"
isDisabled	Опционально. Признак игнорирования регистрации в конфиге, false (по умолчанию) или (true).	true

Для удобства конфигурирования для http движка можно указать один или несколько источников, атрибуты см. Таблица 6.

Таблица 6 Атрибуты объекта подключения к источнику для движка http

Атрибут	Описание	Пример значения
host	IP адрес или полное доменное название сервера СУБД	"192.168.10.21"
port	порт подключения к СУБД	5432

Фрагмент конфига съёмников для одного подключения к СУБД Greenplum с настройками работы съёмников по умолчанию приведён в примере ниже:

```
{
  "http": {
    "gp6-pxf-health": {
      "connection": { "host": "pxf-master", "port": 5888 },
      "additionalData": {
        "cluster_id": "gp6-id"
      }
    }
  }
}
```

```

    },
    "extractors": "pxfMainHealth"
  },
  "gp6-pxf-segments": {
    "connection": [
      { "host": "pxf-segment-1", "port": 5888 },
      { "host": "pxf-segment-2", "port": 5888 }
    ],
    "additionalData": {
      "cluster_id": "gp6-id"
    },
    "extractors": "pxfPrometheus"
  }
}
}

```

3.2.1 Съёмник "Состояние PXF"

В настройках ID съёмника - rxfMainHealth.

Снимать данные о состоянии PXF кластера с мастер-сервера PXF - строковый признак статуса кластера в целом. Регулярность снятия - раз в минуту.

Параметры пользовательской настройки съёмника rxfMainHealth см. Таблица 7.

Таблица 7 пользовательские настройки съёмника rxfMainHealth

Атрибут	Описание	Пример значения
cronUpdateRate	Расписание работы съёмника	"1 * * * *"
protocol	Протокол, http (по умолчанию) или https	"https"
path	Путь (составляющая URL)	"/api-v2/"
method	Метод http запроса	"get"
timeout	Размер "таймаута" при http обращении в мс.	10000

3.2.2 Съёмник "Метрики PXF"

В настройках ID съёмника - rxfPrometheus.

Снимает значения метрик мониторинга PXF с группы хостов (по умолчанию - сегментов). Регулярность снятия - раз в минуту.

Параметры пользовательских настроек съёмника rxfPrometheus см. Таблица 8.

Таблица 8 пользовательские настройки съёмника rxfPrometheus

Атрибут	Описание	Пример значения
cronUpdateRate	Расписание работы съёмника	"1 * * * *"
filter	Сохраняемые метрики с учётом значений тегов, массив строк и/или объектов. Должны принимать варианты значений: строка с названием или маской названия метрики (в качестве "другие символы" указывать символ *) объект с атрибутами: названием или маской названия метрики, маской строки тегов	["*"] или ["pxf_records_sent_total", "pxf_bytes_sent_total"]

Атрибут	Описание	Пример значения
	Символ двойной кавычки для строки значений тегов указывается как \" (с маскированием спец. символа)	или ["pxf*_sent_total"] или [{ "value": "pxf_records_sent_*", "tag": "*segment=\"0\"*" }, { "value": "pxf_records_sent_*", "tag": "*segment=\"2\"*" }, { "value": "pxf_bytes_sent_total", "tag": "*segment=\"5\"*" }, { "value": "pxf_bytes_sent_total", "tag": "*segment=\"6\"*" }]
protocol	Протокол, http (по умолчанию) или https	"https"
path	Путь (составляющая URL)	"/actuator/prometheus"
method	Метод http запроса	"get"
timeout	Размер "таймаута" при http обращении в мс.	10000

3.3 Уровень журналирования (логирования)

Для целей диагностики работы можно указать конкретный уровень детализации сообщений.

traceLevel	Уровень логирования: 0 - INFO - средний уровень детализации, выводится часть уведомлений дополнительно к 0 уровню 1 - ERROR - минимальный уровень, выводятся только ошибки и общие уведомления 2 - TRACE - максимальный уровень детализации, выводится детальная информация для трассировки дополнительно к 1 уровню	1
------------	---	---

Фрагмент конфига:

```
{
  "traceLevel": 1
}
```

3.4 Реквизиты подключения к БД Продукта

Собираемые сервером приложений данные сохраняются в БД, реквизиты подключения к БД формируются автоматически, описаны в Таблица 9

Таблица 9 Параметры конфига-файла для подключения к БД Продукта

Атрибут	Описание	Пример значения
mstoolDb.port	Порт подключения к СУБД Postgres / Citus. Обязательный для настройки	5432
mstoolDb.host	Хост подключения к СУБД Postgres / Citus. Обязательный для настройки	"postgres.ti-mstool"
mstoolDb.db	БД подключения к СУБД Postgres / Citus. Обязательный для настройки	"ti_mstool_db"
mstoolDb.user	Учётная запись пользователя СУБД Postgres / Citus. Обязательный для настройки	"ti_owner"
mstoolDb.password	Пароль подключения к СУБД Postgres / Citus, в зашифрованном виде. Обязательный для настройки, см. п. 2.2.4	"fe...c8"

Фрагмент конфига:

```
{
  "mstoolDb": {
    "port": 5432,
    "host": "postgres.ti-mstool",
    "db": "ti_mstool_db",
    "user": "ti_owner",
    "password": "7859...490a"
  }
}
```

4 Эксплуатационные операции

4.1.1 Журнал работы сервера приложений (nodejs)

Копия журнала сервера приложений из консоли (контейнера nodejs.ms-tool) доступа на хосте, размещается в файле `./mnt/nodejs/logs/nodejs-stdout-stderr.log`.

Мониторить работу сервера приложений удобно с просмотром вывода содержания файла в консоль (ниже команда для linux для вызова из каталога установки продукта):

```
sudo tail -f -n 20 ./mnt/nodejs/logs/nodejs-stdout-stderr.log
```

4.1.2 Определение реквизитов подключения к наблюдаемому кластеру Greenplum

Для определения реквизитов доступа к наблюдаемому кластеру Greenplum необходимо указать актуальные значения в конфиг-файле приложения, в атрибуте группы `greenplumbконфиг-файла` (по умолчанию это файл `./configs/config.json`, в контейнере он мэпится как `/usr/app/resources/configuration/mstool.json`).

В атрибуте `greenplumbуказывается` перечень наблюдаемых кластеров, как минимум, один. Один кластер описывается одним объектом с указанием реквизитов подключения, идентификаторами кластера в накапливаемых данных, и регулярностью работы.

После сохранения изменённого файла параметров при его открытии в режиме редактирования - приложение автоматически отследит изменения и выполнит реинициализацию¹.

В случае если конфиг-файл был перезаписан другим (скопирован), для учёта изменений требуется рестарт контейнера `nodejs.ti-mstool`. Критерием того что новые настройки были приняты приложением является вывод соответствующего сообщения в консоль (контейнера) и в лог-файл приложения (`./mnt/nodejs/logs`)

Описание содержимого конфиг-файла приведено в п. 3

4.1.3 Обновление и проверка лицензии Продукта

Актуальная лицензия должна быть размещена в файле с названием `./configs/license.json` (в контейнере этот файл подключается в контейнер в `/usr/app/resources/license/license.json`). Единственно активной у инсталляции продукта может быть только одна версия лицензии (один файл).

В случае отсутствия валидной лицензии функции Продукта по снятию и трансформации данных недоступны. Ограничения лицензии указаны в файле лицензии в человеко-читаемом виде и удостоверяются подписью.

Проверить наличие корректно настроенной валидной лицензии и наличие в ней ограничений можно следующим вызовом:

```
sudo ./ti-mstool-linux.sh license
```

либо вызовом команды `docker`:

¹ За исключение MacOS. В MacOS необходимым выполнить остановку / старт приложения (как минимум контейнера сервера приложений `nodejs.ti-mstool`)

```
sudo docker exec -it nodejs.ti-mstool sh -c "node ./build/app.to-bin.js --check-license"
```

В случае отсутствия лицензии или ограничения времени её действия будут выведены соответствующие сообщения.

Пример ответа о действующей лицензии

```
[2024.06.05 12:53:19] [Лицензия] Лицензия валидна
```

Пример ответа об отсутствии валидной лицензии, с уточнением причины (в примере - подпись не валидна)

```
[2024.06.05 12:52:44] [Лицензия] Лицензия не действительна. Причина - License sign is corrupted
```

Если лицензия не валидная - контейнер приложения в docker-compose не останавливает работу, но работа приложения останавливается.

4.1.4 Переустановка дашборда

Установка дашборда вручную выполняется штатным методом публикации дашборда в WebUI Grafana. Для этого под учётной записью Grafana с административными правами необходимо выполнить следующие шаги:

- находясь на стартовой странице Grafana в меню, выбрать "Dashboards"
- на странице "Dashboards" после нажатия на кнопку "New" в правом верхнем углу страницы, в выпадающем списке выбрать "Import"
- нажав левой кнопкой мыши на "Upload dashboard JSON file" выбрать импортируемый файл, например "MS Tool resource groups.json". Либо перенести этот файл методом "drag and drop" из файлового приложения в область страницы с указанным названием.
- по необходимости изменить название дашборда (в поле Name) и его ID (в полях Unique identifier (UID))
- в выпадающем списке "PostgreSQL" выбрать название "PostgreSQL" в качестве зарегистрированного в Grafana источника данных

4.1.5 Зачистка дискового пространства от логов / старых данных Продукта

В работе продукта для визуализации используются данные из таблиц витрин.

Более 65% дискового пространства занимают исходные данные съёмника, на основе которых построены витрины. В случае если детальные данные не требуются (штатно они трансформируются в витрины сразу после снятия), устаревшие данные из таблицы `ti_mstool.tel_resource_group` удалять (по значению времени в поле `ts`) от имени пользователя `ti_owner`.

В случае отсутствия необходимости в данных в графиках - зачистку можно провести для всех таблиц `ti_mstool.dm_rg_*`.

Журналы работы (лог файлы) размещаются в подкаталогах `./mnt` - могут быть удалены в любой момент при отсутствии необходимости в их наличии.

Таблицы сырых данных и съёмников реализованы с использованием партиционирования - для удаления данных рекомендуется выполнение зачистки партиции целиком (операция `truncate`).

Таблицы сырых данных и витрин реализованы с колоночной организацией, для которых недоступны операции изменения строк (удаления и обновления). В случае необходимости удаления строк таблицу необходимо перевести из колоночной в строчную организацию (на примере таблицы "table1" в схеме "schema1"):

```
alter_table_set_access_method('schema1.table1', 'heap');
```

После окончания работы процедуры станет доступно удаление и изменение строк в указанной таблице. После выполнения удаления строк рекомендуется выполнить VACUUM для данной таблицы и вернуть её в колоночную организацию хранения:

```
alter_table_set_access_method('schema1.table1', 'columnar');
```

В случае необходимости удаления или изменения строк в партиционированной таблице - данную операцию необходимо выполнить с конкретной партицией.

Перечень партиций с колоночной организацией можно получить в частности следующим запросом:

```
select *
from columnar.options
where relation::text like 'schema1.table1%'
;
```

4.1.6 Переустановка Продукта

Переустановка продукта версии MVP1 выполняется с остановкой старой инсталляции, потерей накопленных данных и влечёт недоступность инсталляции Продукта для конечных пользователей.

Необходимо выполнить следующие действия:

- Распаковать новый дистрибутив в новый каталог
- Скопировать файлы лицензии и конфигурационный файл из предыдущей инсталляции
- Выполнить резервное копирование данных БД, по необходимости - экспорт дашбордов
- Выполнить удаление продукта, см. п. 4.1.7
- Выполнить процедуру установки новой версии (из каталога нового дистрибутива)
- Импортировать резервную копию в СУБД PostgreSQL новой инсталляции (при условии обратной совместимости или наличии процедуры загрузки истории в структуру нового дистрибутива)
- По необходимости импортировать дашборды в Grafana (при условии совместимости со структурой витрин данных в нового дистрибутива)

Пример скрипта выполнения действий см. ниже, при следующих условиях:

- установки продукта выполняются в каталог /opt/{версия продукта}/ti-mstool
- версия предыдущая (установленная) указывается в OLD_VERSION (в примере ниже ti-mstool-0.0.0-1);
- версия новая (устанавливаемая) указывается в VERSION (в примере ниже ti-mstool-0.0.0-2);

- старая (текущая установленная версия) находится в активном состоянии и продолжает работать;
- новая версия уже распакована, но не инсталлирована.

Первая часть - снятие резервной копии и установка новой версии

```
# Старая и новая версии
export OLD_VERSION=ti-mstool-0.0.0-1
export VERSION=ti-mstool-0.0.0-2

# Снятие бэкапа и остановка / удаление имиджей старой версии
cd /opt/${OLD_VERSION}/ti-mstool
sudo docker exec postgres.ti-mstool bash -c "pg_dump -U ti_owner ti_mstool_db >
/var/lib/postgresql/data/ti_mstool_db_backup"
sudo ls -l /mnt/postgres/data/ti_mstool_db_backup
sudo ./ti-mstool-linux.sh remove

# Установка новой версии
cd /opt/${VERSION}/ti-mstool
sudo ./ti-mstool-linux.sh install
```

Необходимо время на выполнении инициализации БД новой версии. Спустя минуту проверить в логе контейнера Postgres окончание инициализации и готовность СУБД к подключению пользователей. Например, командой вывода лога контейнера

```
sudo docker logs -f -n 150 postgres.ti-mstool
```

Признак завершения инициализации - наличие следующей строки (при условии отсутствия сообщений об ошибках):

```
... LOG: database system is ready to accept connections
```

Вторая часть - загрузка резервной копии, копирование лицензии и конфига

```
# Накат из бэкапа старой БД
sudo cp -r /opt/${OLD_VERSION}/ti-mstool/mnt/postgres/data/ti_mstool_db_backup
/opt/${VERSION}/ti-mstool/mnt/postgres/data/
sudo docker exec postgres.ti-mstool bash -c "ls -l
/var/lib/postgresql/data/ti_mstool_db_backup"
sudo docker exec postgres.ti-mstool bash -c "psql -U ti_owner ti_mstool_db <
/var/lib/postgresql/data/ti_mstool_db_backup"

# Копирование содержимого конфига и лицензии
sudo cat /opt/${OLD_VERSION}/ti-mstool/configs/config.json > /opt/${VERSION}/ti-
mstool/configs/config.json
sudo cat /opt/${OLD_VERSION}/ti-mstool/configs/license.json > /opt/${VERSION}/ti-
mstool/configs/license.json
```

4.1.7 Деинсталляция (удаление) продукта

Для деинсталляции последовательно выполнить шаги:

- По необходимости экспортировать дашборды Grafana (если создавались новые или вносились изменения)
- Удалить контейнеры, имиджи и сеть Продукта

```
sudo ./ti-mstool-linux.sh remove
```

- По необходимости сохранить резервную копию данных PostgreSQL с накопленной информацией (из каталога /mnt/postgres/data)
- Удалить каталог продукта, пример для Ubuntu 22 и дистрибутива, развёрнутого в /opt/ti-mstool-0.0.0-0:

```
sudo rm -r /opt/ti-mstool-0.0.0-0
```

Альтернативный способ удаления контейнеров, имиджей и сети - выполнить прямые вызовы docker (версии удаляемых имиджей могут меняться, значения зависят от установленной версии Продукта):

```
# Остановка и удаление контейнеров
sudo docker-compose -f ./docker/dc-app-server.yml down
# Удаление сети
sudo docker network rm ti-mstool
# Удаление имиджей
sudo docker rmi postgres:16.2 grafana:10.4.1 mstool-nodejs:X.X.X-X
```

4.1.8 Резервное копирование и восстановление

Для резервного копирования данных съёмника необходимо вызвать создание резервной копии БД PostgreSQL в файл:

```
sudo docker exec postgres.ti-mstool bash -c "pg_dump -U ti_owner ti_mstool_db > /var/lib/postgresql/data/ti_mstool_db_backup"
```

По результатам выполнения формируется файл, доступный на хосте по имени `./mnt/postgres/data/ti_mstool_db_backup` (относительно каталога развёртывания продукта).

Для восстановления из резервной копии необходимо файл резервной копии разместить по такому же пути относительно каталога развёртывания новой версии.

Восстановление / импорт данных продукта должно выполняться в БД при условии обратной совместимости структуры наборов данных (при этом существующие в БД наборы данных и пользовательские функции не пересоздаются). Может быть выполнено следующим вызовом:

```
sudo docker exec postgres.ti-mstool bash -c "psql -U ti_owner ti_mstool_db < /var/lib/postgresql/data/ti_mstool_db_backup"
```

4.1.9 Изменение порта доступа к WebUI (Grafana)

WebUI по умолчанию доступен на порту 3000.

Для изменения порта необходимо указано новое значение в файле `./docker/dc-app-server.yml`, в ветви `grafana.ti-mstool -> ports`. Для примера ниже фрагмент этого файла с значением порта 3210 вместо 3000:

```
services:
  grafana.ti-mstool:
    ports:
      - '3210:3000'
```

Для применения настроек необходимо рестартовать контейнер `grafana.ti-mstool` или продукт целиком.

4.1.10 Подготовка строки пароля для конфиг-файла

Пароли в конфиг-файле сервера приложений хранятся в зашифрованном виде.

Для получения этой формы пароля из значения в открытом виде необходимо воспользоваться следующим вызовом (для примера значение пароля указано как `ti_owner_password`, это значение необходимо заменить на значение пароля):

```
sudo docker exec nodejs.ti-mstool node ./build/app.to-bin.js --password ti_owner_password
```

! Вызов работает только при работающем приложении (контейнер `nodejs.ti-mstool` должен находиться в выполняемом состоянии).

Пароль в зашифрованном виде будет выведен в консоль после строки "Encrypted password:", его необходимо (в обрамлении кавычек, как строковое значение формата JSON) указать в конфиг-файле.

! Для снижения риска взлома алгоритма, операции подготовки пароля выдают различные результаты, которые при этом содержат одну и ту же корректную строку пароля.

Пример результата выполнения операции подготовки строки пароля - вывод в консоль:

```
Original password: ti_owner_password
Encrypted password:
fe2520a405d93c9c64d5c68dc72e3c97351f0195fea6a954170824f41daf139ef08eef62fd0ea25daa9ff0ceab602
868f571092ee074f2ba9415fc374bdce2a7ab9d038487
```

4.1.11 Получение версии сервера приложений (контейнера nodejs)

Версию приложения можно уточнить следующей командой скрипта управления (из каталога развёртывания актуальной версии):

```
sudo ./ti-mstool-linux.sh version
```

Информация о версии:

```
Информация о версии съёмника данных
0.1.0-1
```

! Версия сервера приложений (контейнера nodejs.ti-mstool) ведётся независимо от версии дистрибутива, они не обязаны совпадать!

5 Нештатные ситуации

В разделе приведены описания типовых проблем в работе Продукта, их диагностике и методам исправления.

5.1 Не запускается Приложение

На компьютере установки и запуска Продукта (далее хост) должны быть установлены продукты docker и docker-compose.

Возможные причины и методы их решения:

Возможная причина	Метод диагностики и исправления
При выполнении инсталляции и/или других команд выдаются ошибки docker	В случае запуска под Windows если не стартован Docker Desktop, при попытке обращения к docker будут выдаваться ошибки с подобным текстом:
	<pre>error during connect: this error may indicate that the docker daemon is not running: Post "http://%2F%2F.%2Fpipe%2Fdocker_engine/v1.24/networks/creat e": open //./pipe/docker_engine: The system cannot find the file specified. 'true' is not recognized as an internal or external command, operable program or batch file.</pre>
	Для исправления проблемы необходимо выполнить запуск Docker Desktop, и повторить операцию.
При запуске выдаётся сообщение об ошибке подключения к PostgreSQL Продукта	В логе работы сервера приложений выдаётся сообщение подобное следующему:
	<pre>[2024.07.24 19:26:37] Failed on initialization:: Error: Mstool DB connect failed.</pre>
	<p>Проблема с паролем для пользователя PostgreSQL.</p> <p>Штатно корректные логин и пароль в зашифрованном виде уже прописаны в конфиге. Для указания актуального логина и пароля необходимо их указать в конфиге, при этом пароль необходимо предварительно подготовить, см. 4.1.10</p>

5.2 Не появляются данные в таблицах с показателями работы Greenplum

Суть проблемы: в таблице с "сырыми" данными СУБД PostgreSQL по утилизации ресурсных групп не появляются свежие данные.

Проверка наличия проблемы - выполнить запрос от имени тех. пользователя PostgreSQL ti_owner в БД ti_mstool_db , в результатах запроса отсутствию записи со свежими значениями времени в атрибуте ts (с указанием вместо ti_mstool.tel... набора данных конкретного типа SQL съёмника - ti_mstool.tel_resource_group или ti_mstool.tel_data_temperature):

```
select * from ti_mstool.tel_... trg
order by ts desc;
```

Возможные причины и методы их решения:

Возможная причина	Метод диагностики и исправления
Контейнеры стартованы без необходимых прав, порты из	На linux и macos по умолчанию необходимо запускать контейнеры с использованием sudo (или из-под root пользователя). В ином случае при

Возможная причина	Метод диагностики и исправления
контейнеров не "пробрасываются" наружу и не видны	<p>запуске без sudo может оказаться что контейнеры по всем признакам работают, но подключиться к ним и между ними не получается.</p> <p>Для диагностика наличия проблемы:</p> <p>Подключиться на порт 10432 к PostgreSQL не получается, при этом к Grafana доступ на порт 3000 есть.</p> <p>Проверить что контейнеры работают - например через "docker stats".</p> <p>Проверить что порт postgres.ti-mstool настроен и должен быть доступен - например через "docker inspect postgres.ti-mstool", в параметрах контейнера должны присутствовать</p> <pre>"NetworkSettings": { "Ports": { "5432/tcp": [{ "HostIp": "0.0.0.0", "HostPort": "10432" }] }, }</pre>
Для данных съёмника РГ - СУБД Greenplum не настроена для работы с ресурсными группами (работает по умолчанию с ресурсными очередями)	<p>Подключиться к СУБД Greenplum от имени учётной записи тех. пользователя Продукта, и выполнить запрос к таблице системного справочника .</p> <p>При наличии причины будет получена ошибка доступа при выполнении запроса.</p>
Для данных съёмника температуры данных - в СУБД Greenplum отключено обновление системных счётчиков доступа к таблицам (включено по умолчанию)	<p>Подключиться к СУБД Greenplum от имени учётной записи тех. пользователя Продукта, и проверить значение настройки.</p> <pre>show track_counts ;</pre> <p>Корректное значение "on". Изменение настройки доступно только суперпользователю.</p> <p>Проверка отсутствия / исправления проблемы - выполнить запрос ниже и зафиксировать значения полученных данных по конкретной таблице, выполнить запрос к этой таблице (например select count(1) from ...), спустя минуту повторно выполнить запрос ниже и сравнить значения счётчиков: запрос должен привести к изменению счётчиков сканирования</p> <pre>select * from gp_dist_random('pg_stat_user_tables')</pre>
Нет сетевого доступа к СУБД Greenplum от SQL съёмника	<p>Подключение к СУБД Greenplum от имени учётной записи тех. пользователя Продукта завершается ошибкой недоступности сервера.</p> <p>Необходимо проверить наличие сетевого доступа до сервера, корректность имени сервера кластера для подключения.</p>
Нет прав у пользователя MS Tool в СУБД Greenplum на выборку из 2х системных таблиц согласно инструкции установки	<p>Подключиться к СУБД Greenplum от имени учётной записи тех. пользователя Продукта, и выполнить запрос к системному справочнику.</p> <p>При наличии причины будет получена ошибка доступа при выполнении запроса.Р</p>
Недостаточно ресурсов для работы пользователя TI MS Tool в СУБД Greenplum	<p>Запросы на запись данных в таблицу съёмника выполняются более 30 секунд.</p> <p>Возможно, высокая сторонняя нагрузка на СУБД PostgreSQL - включая со стороны пользователей Grafana. Рекомендуется выделение БОльшого кол-ва CPU и RAM контейнеру PostgreSQL.</p> <p>Возможно, значительно нагружен хост, недостаточно выделяется CPU и RAM. Рекомендуется снизить нагрузку на хост от сторонних активностей, либо перенести продукт на другой хост.</p>

Возможная причина	Метод диагностики и исправления
	<p>Возможно, накоплено много данных в БД, требуется удаление устаревших данных, выполнение VACUUM на таблицах.</p> <p>Возможно сильно нагружена файловая система хоста, или дисковое пространство выделяемое контейнеру PostgreSQL заполнено более критического уровня, что приводит к деградации скорости выполнения операций записи на диск.</p>
<p>Не корректно прописан доступ к СУБД Greenplum источника в конфиге SQL съёмника</p>	<p>Подключиться к СУБД Greenplum от имени учётной записи тех. пользователя Продукта, с реквизитами из конфиг-файла (параметры группы CONNECTION).</p> <p>При ошибке подключения - тех. пользователь не валидный, проверить корректность указанных реквизитов (сервер, порт, БД, учётная запись с паролем).</p> <p>В случае если реквизиты корректные - проверять сетевую доступность хоста кластера с сервера развёртывания продукта и контейнера сервера приложений.</p>
<p>В логе работы SQL съёмника присутствуют ошибки, включая отсутствие валидной лицензии</p>	<p>В случае сообщений группы "[Лицензия]" см. ошибки отсутствия лицензии.</p>
<p>Ошибка отсутствия лицензии - необходимо выпустить и разместить валидную лицензию</p>	<p>В лог-файле сервера приложений присутствуют сообщения об ошибках, связанных с отсутствием лицензии или несоответствие лицензии.</p> <p>Необходимо проверить параметры лицензии:</p> <ul style="list-style-type: none"> • время окончания действия лицензии больше, чем текущее время; • кол-во сегментов наблюдаемого кластера не превышает ограничения лицензии (при наличии ограничения на кол-во сегментов кластера); • кол-во ресурсных групп наблюдаемого кластера не превышает ограничения лицензии (при наличии ограничения на кол-во ресурсных групп кластера); • название или IP адрес подключения к кластеру Greenplum не соответствует ограничению лицензии (при наличии ограничения на хосты кластера); <p>В случае необходимости обновления лицензии (продления или расширения или снятия ограничений) необходимо обратиться в поддержку Продукта.</p>
<p>Не работает PostgreSQL (ошибки запуска, недостаточно ресурсов CPU/RAM, недостаточно дискового пространства хостовой машины на запись)</p>	<p>Возможные причины ошибки запуска:</p> <ul style="list-style-type: none"> • занят порт хоста. Необходимо изменить порт мэпинга в файле docker-compose • отсутствует докер-имидж для контейнера. Рекомендуется выполнить инициализирующий (первый) запуск через скрипт. • зарегистрирован другой контейнер с тем же именем • недостаточно ресурсов на хосте • запуск выполняется не из-под root или sudo
<p>Не корректно прописан доступ к PostgreSQL (для дефолтовой конфигурации установки должен быть корректным "по умолчанию")</p>	<p>Обратитесь к поддержке. По умолчанию прописывается хост внутренней сети docker.</p>
<p>Нет прав у тех. пользователя для вставки данных в таблицу SQL съёмника РГ (для дефолтовой</p>	<p>Проявляется ошибка при выполнении процедуры обновления витрин на шаге вставки данных.</p>

Возможная причина	Метод диагностики и исправления
конфигурации установки должен быть корректным ("по умолчанию")	Необходимо выдать права на запись на таблицы ti_mstool.reg_* в БД ti_mstool_db для тех. пользователя ti_owner СУБД PostgreSQL. В случае воспроизведения ошибок обратитесь к поддержке Продукта.

5.3 Не появляются данные в таблицах витрин данных

Суть проблемы: в таблицах с данными витрины в СУБД PostgreSQL по утилизации ресурсных групп не появляются свежие данные.

Проверка наличия проблемы - выполнить запрос от имени тех. пользователя PostgreSQL ti_owner в БД ri_mstool_db , в результатах запроса отсутствию записи со свежими значениями времени в атрибуте ts , например:

```
select * from ti_mstool.dm_rg_ram
order by ts desc;
```

Возможные причины и методы их решения:

Возможная причина	Метод диагностики и исправления
Ошибки вызова обновления витрин в конце итерации снятия телеметрии	Включить трассировку, по логу - проверить что снятие выполняется, и формируется корректный вызов в PostgreSQL (с корректным значением ID кластера и времени) который выполняется без ошибок
Данные съёмника регистрируются на отчётные моменты не кратные минуте (с секундами / миллисекундами)	Значения данных съёмника в атрибуте ts содержат значения секунд и/или миллисекунд, отличные от 0. Обратитесь к поддержке Продукта.
Нет данных в таблице съёмника	см. проблемы с отсутствием данных съёмника
У системного пользователя в PostgreSQL нет прав выполнения процедуры / записи в таблицы витрин	Обратитесь к поддержке Продукта.

5.4 Нет данных в графиках Grafana

Суть проблемы: Grafana работает, но данные в графиках дашборда отсутствуют.

Проверка наличия проблемы - залогиниться в Grafana и открыть дашборд, выбрать целевой временной период данных и ID кластера.

Возможные причины и методы их решения:

Возможная причина	Метод диагностики и исправления
Ошибка подключения к PostgreSQL	Зайти под административной учётной записью в Grafana и открыть описание коннектора PostgreSQL. Если при вызове "Test" выдаётся ошибка - проверить работоспособность контейнера СУБД PostgreSQL и корректность указанных реквизитов для подключения к ней.
Нет данных в витрине за выбранный период времени с наложенными фильтрами (вместо данных сообщение "No data in response" или "No data")	При наложенных фильтрах - возможно нет данных по выбранной комбинации ресурсных групп, за выбранные дни недели или часы суток. Проверьте наличие данных при отключении значений фильтров (выбрать в этих фильтрах значение - "All" в англоязычном интерфейсе). Возможно нет данных - см. не появляются данные в витринах.

Возможная причина	Метод диагностики и исправления
Нет данных по выбранному кластеру в таблицах витрин	<p>Не появляются данные в витринах по целевому кластеру по окончании итерации работы съёмника.</p> <p>см. не появляются данные в витринах.</p>
В списке ID кластеров Greenplum нет ожидаемых идентификаторов	<p>Выбрана шкала времени, в рамках которой нет данных в витрине по ожидаемым ID кластеров.</p> <p>ID кластеров и ресурсных групп в фильтрах обновляются согласно наличию данных за выбранный период.</p> <p>см. не появляются данные в витринах.</p> <p>Либо не корректно прописан идентификатор кластера в конфиге продукта, см. 4.1.1.</p>
После изменения конфиг-файла или лицензии - приложение не реагирует на изменения	<p>Приложение отслеживает изменение файла и пересчитывает его после сохранения изменённой версии.</p> <p>Но если файл на хосте удалялся / переименовывался, и на его месте (с его именем) был создан новый файл - Приложение появление нового файла не отслеживает. После подобного изменения необходимо рестартовать Приложение (достаточно рестарта контейнера nodejs.ms-tool)</p>