**ИНСТРУКЦИЯ ПО УСТАНОВКЕ**

**RT.WAREHOUSE**

2025

# [¶](https://docs.data.rt.ru/%D0%94%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D1%86%D0%B8%D1%8F/RT_WareHouse/Vers_2/RT_WareHouse_Installation_of_services_based_of_the_RT_WareHouse_plugin_2#h-1-%D0%B2%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5)1. Введение

**Примечание:** Непосредственно операции по подготовке к созданию и созданию кластера описаны в документации к RT.ClusterManager. Данные о порядке создания компонент и по распределению сервисов компонент по нодам, описанные  в данном документе, относятся к операциям описанным в [п. 8.4.1 и п. 8.4.2 документа “RT.ClusterManager. Руководство администратора”](https://docs.data.rt.ru/%D0%94%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D1%86%D0%B8%D1%8F/RT_ClusterManager/Vers_2/RT_ClusterManager_Admin_Guide_ver2#:~:text=8.4.1%20%D0%A1%D0%BE%D0%B7%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5%20%D0%BA%D0%BE%D0%BC%D0%BF%D0%BE%D0%BD%D0%B5%D0%BD%D1%82%D0%BE%D0%B2).

**Внимание!**Для отсутствия проблем при расширении кластера (expand) необходимо, чтобы hostname хостов кластера заканчивался на "-число", например "m-1",  “s-12” и т.д..

Возможны разные конфигурация кластеров, создаваемых на основании плагина «RT.WareHouse».

В зависимости от ролей нод, определяется их количество:

Master-node от 1 до 2-х;

Slave-node от 0 до любого приемлемого количества.

Для удобства применения в таблице ниже следующие обозначения:

**m-1, m-2** – Master-node (головной сервер управления slave-нодами),

**s-1, s-2 …**  **s-n** – Slave-node (серверы хранения и обработки данных, управляемые Master-nodes).

В данном описании ноды – сервера (хосты) на базе операционных систем семейства Linux, которые будут включены в кластер.

Режим высокой доступности  – это метод проектирования системы, позволяющий достигать высокий уровень доступности кластера в течение определённого промежутка времени. Включение данного режима повышает отказоустойчивость работы компонентов за счёт увеличения требований к аппаратным ресурсам (увеличение количества серверов в кластере). Достигается путём добавления необходимых сервисов, а также путём дублирования существующего управляемого сервиса.

Режим можно включить бегунком (в окне распределения сервисов по нодам выбранного компонента).

**Важно: При развёртывании стандартного кластера необходимо строго придерживаться порядка установки компонентов, указанного в таблице ниже. В иных случаях существует возможность частично откорректировать распределение.**

Примечание: В плагин RT.WareHouse с версии 2.4.6 в “Компонент” GP была добавлена возможность создавать SWAPFILE. По умолчанию он включен. Выключить его можно зайдя в “Компонент" GP , пройдя в “Конфигурация” можно будет увидеть подраздел “swap\_diskqota". При необходимости можно выключить создание SWAPFILE.

Данная конфигурация добавлена т.к. запуска GP + diskquota на хосте должно быть свободно минимум 5GB ОЗУ + SWAP.

Примечание: В RT.ClusterManager версии 2.Х-ХХ был изменен дизайн и терминология, поменялись местами термины “Компонент” и “Сервис”. Так в RT.ClusterManager версии 1.Х-ХХ: Zookeeper, HDFS … были сервисами, и входящие в них элементы программного обеспечения которые привязывались к нодам  назывались компонентами, а в RT.ClusterManager версии 2.Х-ХХ: Zookeeper, HDFS … стали компонентами, и входящие в них элементы программного обеспечения которые привязываются к нодам  называются сервисами. В данном документе используется терминология RT.ClusterManager версии 2.Х-ХХ.

Примечание: Не устанавливать WH и СМ на один хост где OS RedOs7.3с. Начиная с WH 2.4.4 в процесс установки был добавлен перезапуск хостов, это сделано для поддержки ресурсных групп. Если в компоненте GP у настройки “cgroup\_reboot” не поставить False , то начатая установка перезагрузить хост где установлен WH и CM. Установка WH не будет завершена, а так же CM зависнет в статусе установки приложения.

# ****2. Распределение сервисов по нодам - плагин «RT.WareHouse»****

В следующей таблице представлен порядок распределения по нодам компонентов и их сервисов на основании плагина «RT.WareHouse»:

|  |  |  |  |
| --- | --- | --- | --- |
| ***№*** | ***Компонент*** | ***Назначение*** | ***Распределение сервисов по нодам*** |
| ***С выключенным режимом высокой доступности*** | ***С включенным режимом высокой доступности*** |
| 1 | **GP** | База данных, которая хранит данные и обрабатывает запросы.Проверка статуса - см. [п. 3.1](https://docs.data.rt.ru/%D0%94%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D1%86%D0%B8%D1%8F/RT_WareHouse/Vers_2/RT_WareHouse_Installation_of_services_based_of_the_RT_WareHouse_plugin_2#:~:text=3.1%20%D0%9F%D1%80%D0%BE%D0%B2%D0%B5%D1%80%D0%BA%D0%B0%20%D1%81%D1%82%D0%B0%D1%82%D1%83%D1%81%D0%B0%20GP). | Самая минимальная установка на 1 хост (PXF не будет работать):**master** **–** m-1;**segments –**m-1Обычная установка:**master** **–** m-1;**segments –***s-1 (+*s-2 … s-n*)* | Минимальная установка на 2 хоста (PXF не будет работать):**master** **–** m-1;**standby –** m*-2;***segments –**m*-*1*(*или m-2)*.*Обычная установка:**master** **–** m-1;**standby –** m*-2;***segments –**s*-*1*(+*s-2 … s-n)*.* |
| 2 | **PXF** | Расширение для обмена данными с внешними системами.Проверка статуса - см. [п. 3.2](https://docs.data.rt.ru/%D0%94%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D1%86%D0%B8%D1%8F/RT_WareHouse/Vers_2/RT_WareHouse_Installation_of_services_based_of_the_RT_WareHouse_plugin_2#:~:text=3.2%20%D0%9F%D1%80%D0%BE%D0%B2%D0%B5%D1%80%D0%BA%D0%B0%20%D1%81%D1%82%D0%B0%D1%82%D1%83%D1%81%D0%B0&nbsp;PXF). | Обычная установка:**master –** m-1;**segments –***s-1 (+*s-2 … s-n*)* | Обычная установка:**master –** m-1;**segments –***s-1 (+*s-2 … s-n*)* |

# 3. Проверка статуса компонентов

## 3.1 Проверка статуса ****GP****

Успешность установки можно проверить, зайдя на мастер хост и выполнив команду под пользователем gpadmin:

gpstate

Copy

## 3.2 Проверка статуса ****PXF****

Дождитесь успешной установки и проверьте работоспособность **pxf**следующими шагами на мастер ноде под пользователем gpadmin:

а) Добавьте в файл **/data/master/gpseg-1/pg\_hba.conf** строку:

host all gpadmin samenet trust

Copy

б) Выполните команду:

/usr/local/rt.gpdb/bin/pg\_ctl reload -D /data/master/gpseg-1

Copy

в) Подключитесь к базе

psql -d postgres

Copy

г) Выполните скрипт (в блоке **LOCATION**внутри параметра ***DB\_URL*** необходимо указать сетевое имя мастер хоста):

create extension if not exists pxf;

drop table if exists public.test\_pxf\_int;

create table public.test\_pxf\_int (id int);

drop EXTERNAL TABLE if exists public.test\_pxf\_ext;

CREATE EXTERNAL TABLE public.test\_pxf\_ext

(id int)

LOCATION ('pxf://public.test\_pxf\_int?PROFILE=Jdbc&&JDBC\_DRIVER=org.postgresql.Driver&DB\_URL=jdbc:postgresql://node-1.localdomain:5432/postgres&USER=gpadmin')

FORMAT 'CUSTOM' (FORMATTER='pxfwritable\_import');

insert into public.test\_pxf\_int values (1);

select \* from public.test\_pxf\_ext;

Copy

д) Вывод покажет, что внешняя таблица работает: