



**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
«Digit Смарт-БД»**

Инструкция по установке и эксплуатации ПО

Листов 30

2026

СОДЕРЖАНИЕ

Инструкция по установке и эксплуатации ПО «Digit Смарт-БД»5

1. Общие сведения5
2. Назначение инструкции5
3. Требования к серверной среде5
 - 3.1. Операционная система5
 - 3.2. Требования к Java6
 - 3.3. Требования к аппаратным ресурсам6
 - 3.4. Требования к пользователю ОС6
 - 3.5. Требования к сетевым портам6
4. Подготовка к установке7
 - 4.1. Создание каталога установки7
5. Установка ПО «Digit Смарт-БД»7
 - 5.1. Получение дистрибутива7
 - 5.2. Копирование дистрибутива на сервер7
 - 5.3. Распаковка дистрибутива7
6. Настройка переменных окружения8
7. Первичная конфигурация8
 - 7.1. Основной конфигурационный файл8
 - 7.2. Настройка каталогов8
 - 7.3. Настройка портов9
 - 7.4. Настройка памяти JVM10
 - 7.5. Настройка пользователей11
8. Запуск Digit Смарт-БД в ручном режиме11
9. Проверка работоспособности11
 - 9.1. Проверка процесса11
 - 9.2. Проверка портов12
 - 9.3. Проверка HTTP-интерфейса12
 - 9.4. Проверка подключения через консоль12
10. Установка Digit Смарт-БД как systemd-сервиса12
 - 10.1. Создание unit-файла12
 - 10.2. Перечитывание конфигурации systemd13

- 10.3. Включение автозапуска13
- 10.4. Запуск сервиса13
- 10.5. Проверка статуса13
- 11. Остановка и перезапуск13
 - 11.1. Остановка13
 - 11.2. Перезапуск13
 - 11.3. Проверка статуса после перезапуска13
 - 11.4. Просмотр журналов systemd13
- 12. Разворот базы данных13
 - 12.1. Создание базы данных13
 - 12.2. Проверка списка баз данных14
 - 12.3. Создание пользователей базы данных14
- 13. Настройка безопасности14
 - 13.1. Пароли14
 - 13.2. Сетевой доступ14
 - 13.3. TLS15
- 14. Эксплуатация ПО15
 - 14.1. Ежедневные операции15
 - 14.2. Проверка статуса сервиса15
 - 14.3. Проверка журналов15
 - 14.4. Проверка использования диска15
 - 14.5. Проверка потребления памяти15
- 15. Резервное копирование17
 - 15.1. Общие сведения17
 - 15.2. Команда резервного копирования17
 - 15.3. Резервное копирование локальной базы данных17
 - 15.4. Тип резервного копирования18
 - 15.5. Блокировка записи на время backup19
 - 15.6. Пример shell-скрипта резервного копирования19
 - 15.7. Автоматизация резервного копирования через cron20
 - 15.8. Хранение резервных копий20
 - 15.9. Очистка старых резервных копий21

- 16.1. Общие сведения21
- 16.2. Важное ограничение restore21
- 16.3. Подготовка к восстановлению22
- 16.4. Восстановление базы данных через консоль22
- 16.5. Проверка после восстановления22
- 16.6. Проверка доступности сервиса после восстановления23
- 16.7. Откат после неуспешного восстановления23
- 17.1. Перед резервным копированием23
- 17.2. После резервного копирования24
- 17.3. Перед восстановлением24
- 17.4. После восстановления24
- 17. Обновление версии Digit Смарт-БД25
 - 17.1. Подготовка к обновлению25
 - 17.2. Обновление файлов25
 - 17.3. Проверка после обновления25
- 18. Удаление ПО25
 - 18.1. Остановка сервиса25
 - 18.2. Отключение автозапуска26
 - 18.3. Удаление systemd unit26
 - 18.4. Удаление файлов ПО и данных26
- 19. Типовые проблемы и способы устранения26
 - 19.1. Сервер не запускается26
 - 19.2. Порт занят26
 - 19.3. Нет доступа к web-интерфейсу26
 - 19.4. Ошибка доступа к каталогу данных27
 - 19.5. Недостаточно памяти27
- 20. Минимальный чеклист установки27

ИНСТРУКЦИЯ ПО УСТАНОВКЕ И ЭКСПЛУАТАЦИИ ПО «DIGIT СМАРТ-БД»

1. Общие сведения

Digit Смарт-БД представляет собой серверное программное обеспечение для хранения, обработки и предоставления доступа к данным в объектно-документной модели.

ПО «Digit Смарт-БД» устанавливается на серверную инфраструктуру организации и работает как самостоятельный серверный процесс. Доступ к Digit Смарт-БД осуществляется через сетевые интерфейсы и клиентские инструменты администрирования.

Digit Смарт-БД может использоваться:

1. Для хранения структурированных и полуструктурированных данных.
2. Для работы с документными объектами.
3. Для выполнения запросов к данным.
4. Для администрирования баз данных.
5. Для интеграции с прикладными информационными системами.

2. Назначение инструкции

Настоящая инструкция описывает порядок:

1. Подготовки серверной среды.
2. Установки ПО «Digit Смарт-БД».
3. Первичного разворота ПО.
4. Запуска и остановки сервера.
5. Проверки работоспособности.
6. Базовой эксплуатации.
7. Резервного копирования и восстановления.
8. Диагностики типовых проблем.

3. Требования к серверной среде

3.1. Операционная система

ПО «Digit Смарт-БД» может быть установлено на серверы под управлением операционных систем семейства Linux.

Рекомендуемые ОС:

1. Astra Linux.
2. Debian.
3. Ubuntu Server.
4. Red Hat Enterprise Linux.

5. CentOS.

3.2. Требования к Java

Digit Смарт-БД является Java-приложением. Для запуска требуется установленная Java Runtime Environment или Java Development Kit.

Рекомендуемая версия: Java 8.

Проверка установленной версии Java: `java -version`.

Пример ожидаемого результата: `java version "1.8.0_xxx"` или `openjdk version "1.8.0_xxx"`.

3.3. Требования к аппаратным ресурсам

Минимальные требования для тестовой установки:

Ресурс	Минимальное значение
CPU	2 vCPU
RAM	4 GB
Disk	20 GB
Network	1 Gbit/s

Рекомендуемые требования для промышленной эксплуатации:

Ресурс	Рекомендуемое значение
CPU	4-8 vCPU
RAM	16-32 GB
Disk	SSD/NVMe
Network	1-10 Gbit/s

Размер дискового пространства зависит от объема данных, журналов, индексов и резервных копий.

3.4. Требования к пользователю ОС

Для эксплуатации Digit Смарт-БД рекомендуется создать отдельного системного пользователя.

Пример: `sudo useradd -r -s /sbin/nologin smartdb`

Создание отдельного пользователя снижает риски эксплуатации ПО от имени root.

3.5. Требования к сетевым портам

Для работы Digit Смарт-БД должны быть доступны сетевые порты сервера.

Типовая схема портов, унаследованная от серверной модели Digit Смарт-БД:

Назначение	Порт по умолчанию
Бинарный протокол	2424
HTTP / Web-интерфейс	2480

У Digit Смарт-БД сервер по умолчанию использует порт 2424 для бинарного протокола и 2480 для HTTP-доступа; при занятости портов может использоваться следующий свободный порт из диапазона.

Рекомендуется явно закрепить порты в конфигурации и открыть только необходимые сетевые направления.

4. Подготовка к установке

4.1. Создание каталога установки

Рекомендуемый каталог установки: /opt/smartdb.

Создать каталог: `sudo mkdir -p /opt/smartdb`.

Назначить владельца: `sudo chown -R smartdb:smartdb /opt/smartdb`.

5. Установка ПО «Digit Смарт-БД»

5.1. Получение дистрибутива

Дистрибутив Digit Смарт-БД поставляется в виде архива: smartdb-<version>.tar.gz

Пример: smartdb-1.0.0.tar.gz

Скачать можно с <https://cloud.smart-consulting.ru/s/2Jee5B9CpDdBRdt>

При запросе пароля ввести по запросу.

5.2. Копирование дистрибутива на сервер

Пример копирования: `scp smartdb-1.0.0.tar.gz user@server:/tmp/`

5.3. Распаковка дистрибутива

Перейти на сервер и выполнить: `sudo tar -xzf /tmp/smartdb-1.0.0.tar.gz -C /opt/smartdb --strip-components=1`

Назначить владельца: `sudo chown -R smartdb:smartdb /opt/smartdb`

Проверить структуру каталогов: `ls -la /opt/smartdb`

Ожидаемая структура:

/opt/smartdb

/bin

/config

/databases

/lib
/plugins
/backup
/log

6. Настройка переменных окружения

Рекомендуется задать переменную окружения: SMARTDB_HOME=/opt/smartdb.

Добавить в профиль пользователя или сервисный unit-файл: export SMARTDB_HOME=/opt/smartdb.

Для совместимости с внутренними скриптами может использоваться переменная: JAVA_HOME=/path/to/java.

Проверка:

1. echo \$SMARTDB_HOME
2. echo \$JAVA_HOME

7. Первичная конфигурация

7.1. Основной конфигурационный файл

Основной серверный конфигурационный файл Digit Смарт-БД находится в каталоге: \$SMARTDB_HOME/config/smartdb-server-config.xml

Где: SMARTDB_HOME=/opt/smartdb

Файл smartdb-server-config.xml используется для настройки серверных параметров Digit Смарт-БД, включая сетевые протоколы, обработчики, пользователей, хранилища и другие параметры серверного процесса.

Перед изменением конфигурации рекомендуется создать резервную копию файла: cp \$SMARTDB_HOME/config/ smartdb-server-config.xml \ \$SMARTDB_HOME/config/ smartdb-server-config.xml.backup.\$(date +%F)

После изменения конфигурации требуется перезапуск сервера Digit Смарт-БД.

7.2. Настройка каталогов

Digit Смарт-БД использует каталог databases, расположенный внутри домашнего каталога установки.

База данных всегда работает с каталогом: \$SMARTDB_HOME/databases

Например, если Digit Смарт-БД установлена в: /opt/smartdb, то каталог баз данных будет: /opt/smartdb/databases. Именно в этом каталоге размещаются файлы создаваемых баз данных.

Перед запуском необходимо проверить наличие каталога: `ls -la $SMARTDB_HOME/databases.`

Если каталог отсутствует, его необходимо создать: `mkdir -p $SMARTDB_HOME/databases.`

Назначить права сервисному пользователю: `chown -R smartdb:smartdb $SMARTDB_HOME/databases.`

Проверить права:

1. `ls -la $SMARTDB_HOME.`
2. `ls -la $SMARTDB_HOME/databases.`

7.3. Настройка портов

Сетевые порты Digit Смарт-БД задаются в файле: `$SMARTDB_HOME/config/smartdb-server-config.xml.`

Настройка выполняется в секции сетевых `listeners/protocols.`

Типовой фрагмент конфигурации:

```
<listeners>
  <listener protocol="binary" ip-address="0.0.0.0" port-range="2424-2430"
socket="default"/>
  <listener protocol="http" ip-address="0.0.0.0" port-range="2480-2490" socket="default"/>
</listeners>
```

Где:

Параметр	Назначение
<code>protocol="binary"</code>	Бинарный протокол подключения клиентов
<code>protocol="http"</code>	HTTP-доступ, включая web-интерфейс
<code>ip-address</code>	Адрес, на котором сервер принимает подключения
<code>port-range</code>	Диапазон портов, доступных для listener
<code>socket</code>	Используемая socket-конфигурация

Стандартные порты:

Протокол	Порт
Binary	2424
HTTP	2480

Для промышленной установки рекомендуется задавать не диапазон, а фиксированные значения портов, если это поддерживается текущей версией конфигурации:

```

<listeners>
  <listener protocol="binary" ip-address="0.0.0.0" port-range="2424-2424"
socket="default"/>
  <listener protocol="http" ip-address="0.0.0.0" port-range="2480-2480" socket="default"/>
</listeners>

```

После изменения портов необходимо перезапустить Digit Смарт-БД: `sudo systemctl restart smartdb`.

Проверка открытых портов:

1. `ss -lntp | grep 2424`
2. `ss -lntp | grep 2480`

7.4. Настройка памяти JVM

Память JVM для Digit Смарт-БД настраивается через переменную окружения: `SMARTDB_OPTS_MEMORY`.

Эта переменная используется стартовыми скриптами сервера для передачи параметров памяти в JVM.

Пример для тестовой среды: `export SMARTDB_OPTS_MEMORY="-Xms512m -Xmx2g"`.

Пример для промышленной среды: `export SMARTDB_OPTS_MEMORY="-Xms4g -Xmx4g"` или при большем объеме данных: `export SMARTDB_OPTS_MEMORY="-Xms8g -Xmx8g"`.

Рекомендуется задавать одинаковые значения `-Xms` и `-Xmx` для стабильной работы JVM: `export SMARTDB_OPTS_MEMORY="-Xms4g -Xmx4g"`.

Переменную можно задать в профиле пользователя, но для промышленной эксплуатации правильнее фиксировать ее в `systemd unit-файле`.

Пример:

```

[Service]
Type=simple
User=smartdb
Group=smartdb
Environment=SMARTDB_HOME=/opt/smartdb
Environment= SMARTDB_OPTS_MEMORY=-Xms4g -Xmx4g
WorkingDirectory=/opt/smartdb
ExecStart=/opt/smartdb/bin/server.sh
Restart=on-failure
RestartSec=10

```

LimitNOFILE=65535

После изменения параметров памяти необходимо перезапустить сервис:

```
sudo systemctl daemon-reload
```

```
sudo systemctl restart smartdb
```

Проверить, что параметры применились:

```
ps aux | grep java
```

В выводе должны присутствовать параметры:

```
-Xms4g -Xmx4g
```

7.5. Настройка пользователей

При первичном запуске необходимо задать административного пользователя.

Минимально должны быть определены:

Пользователь	Назначение
admin	Администрирование сервера и БД
reader	Чтение данных
writer	Чтение и изменение данных
service	Сервисный пользователь для интеграций

Пароли пользователей должны храниться в соответствии с политиками безопасности организации.

Не допускается использовать пароли по умолчанию в промышленной среде.

8. Запуск Digit Смарт-БД в ручном режиме

Перейти в каталог установки: `cd /opt/smartdb`

Запустить сервер: `sudo -u smartdb ./bin/server.sh`

Для Windows-поставки может использоваться скрипт: `bin\server.bat`

После запуска необходимо проверить логи: `tail -f $SMARTDB_HOME/log/smartdb.log`

Признаки успешного запуска:

1. Сервер запущен
2. Порты открыты
3. Ошибки и исключения отсутствуют
4. База данных доступна

9. Проверка работоспособности

9.1. Проверка процесса

```
ps aux | grep smartdb
```

или:

```
ps aux | grep java
```

9.2. Проверка портов

```
ss -lntp | grep 2424
```

```
ss -lntp | grep 2480
```

или:

```
netstat -lntp | grep 2424
```

```
netstat -lntp | grep 2480
```

9.3. Проверка HTTP-интерфейса

Открыть в браузере: `http://<server-host>:2480`

Для локальной проверки: `curl -I http://localhost:2480`

Ожидаемый результат: HTTP/1.1 200 OK или другой корректный ответ сервера.

Загрузится стартовая страница Digit Смарт-БД.

9.4. Проверка подключения через консоль

Если в поставке есть консольный клиент: `cd /opt/smartdb/bin./console.sh`

Подключение к серверу: `connect remote:localhost admin <password>`

10. Установка Digit Смарт-БД как systemd-сервиса

10.1. Создание unit-файла

Создать файл: `sudo nano /etc/systemd/system/smartdb.service`

Пример содержимого:

```
[Unit]
```

```
Description=SmartDB Server
```

```
After=network.target
```

```
[Service]
```

```
Type=simple
```

```
User=smartdb
```

```
Group=smartdb
```

```
Environment=SMARTDB_HOME=/opt/smartdb
```

```
WorkingDirectory=/opt/smartdb
```

```
ExecStart=/opt/smartdb/bin/server.sh
```

```
ExecStop=/bin/kill -TERM $MAINPID
```

```
Restart=on-failure
```

RestartSec=10

LimitNOFILE=65535

[Install]

WantedBy=multi-user.target

10.2. Перечитывание конфигурации systemd

sudo systemctl daemon-reload

10.3. Включение автозапуска

sudo systemctl enable smartdb

10.4. Запуск сервиса

sudo systemctl start smartdb

10.5. Проверка статуса

sudo systemctl status smartdb

11. Остановка и перезапуск

11.1. Остановка

sudo systemctl stop smartdb

11.2. Перезапуск

sudo systemctl restart smartdb

11.3. Проверка статуса после перезапуска

sudo systemctl status smartdb

11.4. Просмотр журналов systemd

journalctl -u smartdb -f

12. Разворот базы данных

12.1. Создание базы данных

Создание базы данных выполняется через административную консоль, web-интерфейс или поставляемую утилиту.

Пример через консоль: create database remote:localhost/<database_name> root <password>
plocal

Пример: create database remote:localhost/demo admin <password> plocal

12.2. Проверка списка баз данных

Через консоль: `list databases`, или через web-интерфейс администратора.

12.3. Создание пользователей базы данных

Для каждой базы данных необходимо создать пользователей с минимально необходимыми правами.

Рекомендуемая модель:

Роль	Права
admin	Полное администрирование
reader	Только чтение
writer	Чтение и изменение данных
integration	Доступ для внешних систем

Не рекомендуется использовать административного пользователя в прикладных интеграциях.

13. Настройка безопасности

13.1. Пароли

После установки необходимо:

1. Сменить все пароли по умолчанию.
2. Запретить использование простых паролей.
3. Исключить хранение паролей в открытом виде.
4. Ограничить доступ к конфигурационным файлам.

Права на конфигурационные файлы:

```
sudo chmod 640 /opt/smartdb/config/*
```

```
sudo chown -R smartdb:smartdb /opt/smartdb/config
```

13.2. Сетевой доступ

Доступ к портам Digit Смарт-БД должен быть разрешен только доверенным узлам.

Пример ограничения через firewall:

```
sudo ufw allow from <trusted-ip> to any port 2424
```

```
sudo ufw allow from <trusted-ip> to any port 2480
```

Для промышленной эксплуатации web-интерфейс администратора не должен быть доступен из публичных сетей.

13.3. TLS

Для промышленной эксплуатации рекомендуется включить TLS.

Должны быть определены:

1. Keystore.
2. Truststore.
3. Сертификат сервера.
4. Пароли хранилищ.
5. Разрешенные протоколы TLS.

Минимально допустимые версии:

TLS 1.2

TLS 1.3

14. Эксплуатация ПО

14.1. Ежедневные операции

Администратор должен контролировать:

1. Статус сервиса.
2. Доступность портов.
3. Размер каталогов данных.
4. Размер журналов.
5. Ошибки в логах.
6. Состояние резервных копий.
7. Наличие свободного места на диске.
8. Использование CPU и RAM.

14.2. Проверка статуса сервиса

```
sudo systemctl status smartdb
```

14.3. Проверка журналов

```
journalctl -u smartdb -n 200 или tail -n 200 $SMARTDB_HOME/log/smartdb.log
```

14.4. Проверка использования диска

```
du -sh $SMARTDB_HOME
```

```
df -h
```

14.5. Проверка потребления памяти

```
free -h
```

```
ps aux --sort=-%mem | head
```


15. РЕЗЕРВНОЕ КОПИРОВАНИЕ

15.1. Общие сведения

Digit Смарт-БД поддерживает резервное копирование базы данных в архивный файл формата ZIP.

Резервное копирование выполняется для конкретной базы данных. Имя базы данных указывается в URL базы данных при запуске команды резервного копирования.

Рекомендуемый основной способ резервного копирования для Digit Смарт-БД - выполнение команды на сервере, где установлена Digit Смарт-БД, с использованием локального доступа к каталогу базы данных.

Каталог баз данных Digit Смарт-БД: `$SMARTDB_HOME/databases`

Пример: `/opt/smartdb/databases`

Резервные копии рекомендуется хранить в отдельном каталоге: `$SMARTDB_HOME/backup`

Пример: `/opt/smartdb/backup`

Перед настройкой резервного копирования необходимо создать каталог для backup-файлов:

```
mkdir -p $SMARTDB_HOME/backup
```

```
chown -R smartdb:smartdb $SMARTDB_HOME/backup
```

15.2. Команда резервного копирования

Для резервного копирования используется скрипт: `$SMARTDB_HOME/bin/backup.sh`

Формат команды: `$SMARTDB_HOME/bin/backup.sh <database_url> <user> <password> <backup_file_path> [backup_type]`

Где:

Параметр	Описание
<code><database_url></code>	URL базы данных, которую необходимо сохранить
<code><user></code>	Пользователь базы данных
<code><password></code>	Пароль пользователя
<code><backup_file_path></code>	Путь к создаваемому ZIP-файлу резервной копии
<code>[backup_type]</code>	Необязательный тип резервного копирования

15.3. Резервное копирование локальной базы данных

Для штатного резервного копирования используется локальный URL базы данных: `plocal:$SMARTDB_HOME/databases/<database_name>`

Формат:

```
$SMARTDB_HOME/bin/backup.sh \  
  plocal:$SMARTDB_HOME/databases/<database_name> \  
  <user> \  
  <password> \  
  $SMARTDB_HOME/backup/<database_name>-<date>.zip
```

Пример для базы данных smartdb_demo:

```
$SMARTDB_HOME/bin/backup.sh \  
  plocal:$SMARTDB_HOME/databases/smartdb_demo \  
  admin \  
  admin_password \  
  $SMARTDB_HOME/backup/smartdb_demo-2026-05-13.zip
```

В данном примере выполняется резервное копирование базы данных:

```
$SMARTDB_HOME/databases/smartdb_demo
```

Результат будет сохранен в файл:

```
$SMARTDB_HOME/backup/smartdb_demo-2026-05-13.zip
```

15.4. Тип резервного копирования

По умолчанию используется стандартный тип резервного копирования.

Пример:

```
$SMARTDB_HOME/bin/backup.sh \  
  plocal:$SMARTDB_HOME/databases/smartdb_demo \  
  admin \  
  admin_password \  
  $SMARTDB_HOME/backup/smartdb_demo-2026-05-13.zip
```

Также может использоваться режим lvm, если серверная инфраструктура поддерживает LVM snapshot.

Пример:

```
$SMARTDB_HOME/bin/backup.sh \  
  plocal:$SMARTDB_HOME/databases/smartdb_demo \  
  admin \  
  admin_password \  
  $SMARTDB_HOME/backup/smartdb_demo-2026-05-13.zip \  
  lvm
```

Режим lvm допускается использовать только при выполнении следующих условий:

1. Сервер использует LVM.
2. Каталог базы данных размещен на LVM-томе.
3. Процедура backup/restore проверена на тестовом стенде.
4. Администратор понимает схему snapshot и очистки временных томов.

Если эти условия не выполнены, необходимо использовать стандартный режим резервного копирования.

15.5. Блокировка записи на время backup

При выполнении стандартного резервного копирования база данных может временно блокировать операции записи.

Это означает:

1. Чтение данных может продолжаться.
2. Операции изменения данных могут ожидать завершения backup.
3. При большой базе данных время ожидания может быть заметным.
4. Backup необходимо выполнять в период минимальной нагрузки.

Для промышленной эксплуатации рекомендуется:

1. Запускать backup в технологическое окно.
2. Контролировать время выполнения операции.
3. Контролировать размер создаваемого ZIP-файла.
4. Проверять код завершения команды.
5. Периодически проверять восстановление из резервной копии на отдельном стенде.

15.6. Пример shell-скрипта резервного копирования

Пример скрипта:

```
#!/bin/bash
```

```
set -e
```

```
SMARTDB_HOME=/opt/smartdb
```

```
DATABASE_NAME=smartdb_demo
```

```
DATABASE_USER=admin
```

```
DATABASE_PASSWORD=admin_password
```

```
BACKUP_DIR=$SMARTDB_HOME/backup
```

```
BACKUP_DATE=$(date +%F-%H%M%S)
```

```
BACKUP_FILE=$BACKUP_DIR/${DATABASE_NAME}-${BACKUP_DATE}.zip
```

```
mkdir -p "$BACKUP_DIR"
```

```
$SMARTDB_HOME/bin/backup.sh \  
  plocal:$SMARTDB_HOME/databases/$DATABASE_NAME \  
  $DATABASE_USER \  
  $DATABASE_PASSWORD \  
  $BACKUP_FILE
```

```
echo "Backup completed: $BACKUP_FILE"
```

Файл можно сохранить как:

```
/opt/smartdb/bin/smartdb-backup.sh
```

Назначить права:

```
chmod +x /opt/smartdb/bin/smartdb-backup.sh
```

```
chown smartdb:smartdb /opt/smartdb/bin/smartdb-backup.sh
```

Запуск:

```
sudo -u smartdb /opt/smartdb/bin/smartdb-backup.sh
```

15.7. Автоматизация резервного копирования через cron

Пример ежедневного запуска в 02:00:

```
crontab -e
```

Добавить строку:

```
0 2 * * * /opt/smartdb/bin/smartdb-backup.sh >> /var/log/smartdb/backup.log 2>&1
```

Перед использованием cron необходимо убедиться, что:

1. Скрипт запускается вручную без ошибок.
2. Пользователь имеет права на каталог базы данных.
3. Пользователь имеет права на каталог резервных копий.
4. Лог backup создается и доступен для анализа.

15.8. Хранение резервных копий

Резервные копии не должны храниться только на том же сервере, где размещена рабочая база данных.

Рекомендуется использовать следующую схему:

Тип хранения	Назначение
Локальный backup	Быстрое восстановление после локальной ошибки
Внешнее файловое хранилище	Защита от потери сервера
Долгосрочный архив	Выполнение требований регламента хранения

Минимальный рекомендуемый срок хранения:

Тип копии	Срок хранения
Ежедневные копии	7-14 дней
Еженедельные копии	1-2 месяца
Ежемесячные копии	По регламенту организации

15.9. Очистка старых резервных копий

Пример удаления backup-файлов старше 14 дней: `find $SMARTDB_HOME/backup -name "*.zip" -type f -mtime +14 -delete`

Желательно не выполнять удаление без логирования.

Более безопасный вариант: `find $SMARTDB_HOME/backup -name "*.zip" -type f -mtime +14 -print`

16. Восстановление из резервной копии

16.1. Общие сведения

Восстановление выполняется из ZIP-файла резервной копии.

Пример файла резервной копии: `$SMARTDB_HOME/backup/smartdb_demo-2026-05-13.zip`

Восстановление должно выполняться только администратором Digit Смарт-БД.

Перед восстановлением необходимо:

1. Остановить прикладные сервисы, работающие с базой.
2. Убедиться, что выбран правильный backup-файл.
3. Сохранить текущее состояние каталога базы данных.
4. Проверить наличие свободного места на диске.
5. Зафиксировать время начала восстановления.

16.2. Важное ограничение restore

Операция восстановления не выполняет слияние данных.

Это значит:

1. Restore восстанавливает состояние базы данных из backup-файла.
2. Restore не объединяет старые данные с текущими.
3. Данные, появившиеся после создания backup-файла, не будут автоматически сохранены.
4. Для переноса или объединения данных должен использоваться отдельный регламент экспорта/импорта.

Нельзя использовать restore как механизм “добавить данные из backup в текущую базу”.

16.3. Подготовка к восстановлению

Остановить сервис Digit Смарт-БД: `sudo systemctl stop smartdb`

Проверить, что процесс остановлен:

```
ps aux | grep smartdb
```

```
ps aux | grep java
```

Создать копию текущего каталога базы данных:

```
cp -r \
```

```
SMARTDB_HOME/databases/smartdb_demo \
```

```
SMARTDB_HOME/databases/smartdb_demo.before-restore.$(date +%F-%H%M%S)
```

Проверить наличие backup-файла:

```
ls -lh SMARTDB_HOME/backup/smartdb_demo-2026-05-13.zip
```

16.4. Восстановление базы данных через консоль

Запустить Digit Смарт-БД: `sudo systemctl start smartdb`

Запустить консоль: `cd SMARTDB_HOME/bin./console.sh`

Выполнить восстановление: `RESTORE DATABASE`

```
SMARTDB_HOME/backup/smartdb_demo-2026-05-13.zip
```

После завершения операции выйти из консоли: `EXIT`

16.5. Проверка после восстановления

После восстановления необходимо проверить список баз данных: `LIST DATABASES`

Подключиться к восстановленной базе: `CONNECT`

```
PLOCAL:SMARTDB_HOME/databases/smartdb_demo admin admin_password
```

Выполнить тестовый запрос: `SELECT FROM V LIMIT 1` или другой тестовый запрос, соответствующий структуре конкретной базы данных.

Затем проверить журнал сервера: `journalctl -u smartdb -n 200` или `tail -n 200 SMARTDB_HOME/log/smartdb.log`

16.6. Проверка доступности сервиса после восстановления

Проверить статус сервиса: `sudo systemctl status smartdb`.

Проверить сетевые порты:

```
ss -lntp | grep 2424
```

```
ss -lntp | grep 2480
```

Проверить HTTP-доступ: `curl -I http://localhost:2480`

Проверить подключение прикладного сервиса к базе данных.

16.7. Откат после неуспешного восстановления

Если восстановление завершилось ошибкой, необходимо:

1. Остановить Digit Смарт-БД.
2. Удалить поврежденный или частично восстановленный каталог базы.
3. Вернуть сохраненную копию каталога.
4. Запустить Digit Смарт-БД.
5. Проверить работоспособность.

Пример:

```
sudo systemctl stop smartdb
```

```
rm -rf $SMARTDB_HOME/databases/smartdb_demo
```

```
mv \
```

```
  $SMARTDB_HOME/databases/smartdb_demo.before-restore.2026-05-13-120000 \
```

```
  $SMARTDB_HOME/databases/smartdb_demo
```

```
sudo systemctl start smartdb
```

После запуска проверить статус: `sudo systemctl status smartdb`

17. Контрольный чеклист backup/restore

17.1. Перед резервным копированием

Проверка	Статус
Digit Смарт-БД запущена	
Известно имя базы данных	
Есть пользователь с правами на backup	
Каталог backup существует	

Проверка	Статус
Есть свободное место на диске	
Backup выполняется в период низкой нагрузки	
Старые backup-файлы очищаются по регламенту	

17.2. После резервного копирования

Проверка	Статус
ZIP-файл создан	
Размер файла больше 0	
Команда завершилась без ошибки	
Backup-файл скопирован во внешнее хранилище	
Запись о backup есть в журнале	

17.3. Перед восстановлением

Проверка	Статус
Выбран правильный backup-файл	
Прикладные сервисы остановлены	
Текущий каталог базы сохранен	
Есть свободное место на диске	
Согласовано окно восстановления	
Зафиксирован ответственный администратор	

17.4. После восстановления

Проверка	Статус
База данных восстановлена	
Digit Смарт-БД запускается без ошибок	
Подключение к базе работает	
Тестовые запросы выполняются	
Прикладные сервисы подключаются	
Ошибки в логах отсутствуют	
Результат восстановления зафиксирован	

17. Обновление версии Digit Смарт-БД

17.1. Подготовка к обновлению

Перед обновлением необходимо:

1. Зафиксировать текущую версию.
2. Остановить сервис.
3. Выполнить резервное копирование.
4. Сохранить конфигурационные файлы.
5. Проверить совместимость новой версии.
6. Проверить инструкцию миграции.

17.2. Обновление файлов

Пример:

```
sudo systemctl stop smartdb
```

```
sudo cp -r /opt/smartdb /opt/smartdb.backup.$(date +%F)
```

```
sudo tar -xzf /tmp/smartdb-new-version.tar.gz -C /opt/smartdb --strip-components=1
```

```
sudo chown -R smartdb:smartdb /opt/smartdb
```

```
sudo systemctl start smartdb
```

17.3. Проверка после обновления

После обновления необходимо проверить:

1. Статус сервиса.
2. Версию ПО.
3. Доступность web-интерфейса.
4. Подключение к базам данных.
5. Выполнение запросов.
6. Ошибки в логах.
7. Совместимость клиентских приложений.

18. Удаление ПО

18.1. Остановка сервиса

```
sudo systemctl stop smartdb
```

18.2. Отключение автозапуска

```
sudo systemctl disable smartdb
```

18.3. Удаление systemd unit

```
sudo rm /etc/systemd/system/smartdb.service
```

```
sudo systemctl daemon-reload
```

18.4. Удаление файлов ПО и данных

```
sudo rm -rf /opt/smartdb
```

19. Типовые проблемы и способы устранения

19.1. Сервер не запускается

Проверить:

```
sudo systemctl status smartdb
```

```
journalctl -u smartdb -n 200
```

Возможные причины:

1. Не установлена Java.
2. Неверная версия Java.
3. Ошибка в конфигурации.
4. Недостаточно памяти.
5. Нет прав на каталоги.
6. Занят сетевой порт.
7. Повреждены файлы базы данных.

19.2. Порт занят

Проверить:

```
ss -lntp | grep 2480
```

```
ss -lntp | grep 2424
```

Если порт занят другим процессом, необходимо:

1. Остановить конфликтующий процесс.
2. Изменить порт Digit Смарт-БД.
3. Перезапустить Digit Смарт-БД.

19.3. Нет доступа к web-интерфейсу

Проверить:

1. Запущен ли сервис.
2. Открыт ли порт.

3. Не блокирует ли firewall.
4. Правильно ли указан host.
5. Нет ли bind только на localhost.

19.4. Ошибка доступа к каталогу данных

Проверить владельца:

```
ls -la /var/lib/smartdb
```

Исправить:

```
sudo chown -R smartdb:smartdb /var/lib/smartdb
```

19.5. Недостаточно памяти

Проверить:

```
free -h
```

```
journalctl -u smartdb -n 200
```

Решения:

1. Увеличить RAM.
2. Уменьшить -Xmx.
3. Проверить настройки кэша.
4. Проверить нагрузку.
5. Проверить объем данных и индексов.

20. Минимальный чеклист установки

Перед передачей Digit Смарт-БД в эксплуатацию необходимо проверить:

Проверка	Статус
Java установлена	
Создан пользователь smartdb	
Создан каталог /opt/smardb	
Дистрибутив распакован	
Права на файлы настроены	
Порты настроены явно	
Пароли по умолчанию изменены	
Сервис запускается вручную	
Создан systemd unit	
Автозапуск включен	

Проверка	Статус
Web-интерфейс доступен	
Консольное подключение работает	
Создана тестовая база данных	
Выполнен тестовый запрос	
Настроено резервное копирование	
Проверены журналы	
Ошибки отсутствуют	