GlobalERP

СИСТЕМА УПРАВЛЕНИЯ ПРЕДПРИЯТИЕМ
И ПЛАТФОРМА ДЛЯ ЦИФРОВИЗАЦИИ
БИЗНЕС-ПРОЦЕССОВ

Руководство по развертыванию Global3 на базе Oracle

© Global ERP

v0.2.3 03.11.2026

Содержание

Категория документа:

1 Установка и администрирование Oracle **3**
1.1 Установка Oracle Database . . . 3
1.2 Настройка listener . . . 7
1.3 Создание базы данных . . . 10
1.4 Экспорт схемы с помощью expdp . . . 19
1.5 Импорт схемы с помощью impdp . . . 21

2 Установка и настройка сервера Global3 **24**
2.1 Необходимое программное обеспечение . . . 25
2.2 Установка Oracle Client 11 . . . 25
2.3 Развертывание сервера приложений GlobalServer . . . 27

3 Обновление сервера приложений **31**
3.1 Подготовка окружения . . . 31
3.2 Подготовка скрипта config.sh . . . 32
3.3 Настройка прав на скрипты обновления . . . 32
3.4 Запуск обновления . . . 32
3.5 Пример запуска обновления . . . 32
3.6 Проверка parameters.sh . . . 34

1 Установка и администрирование Oracle

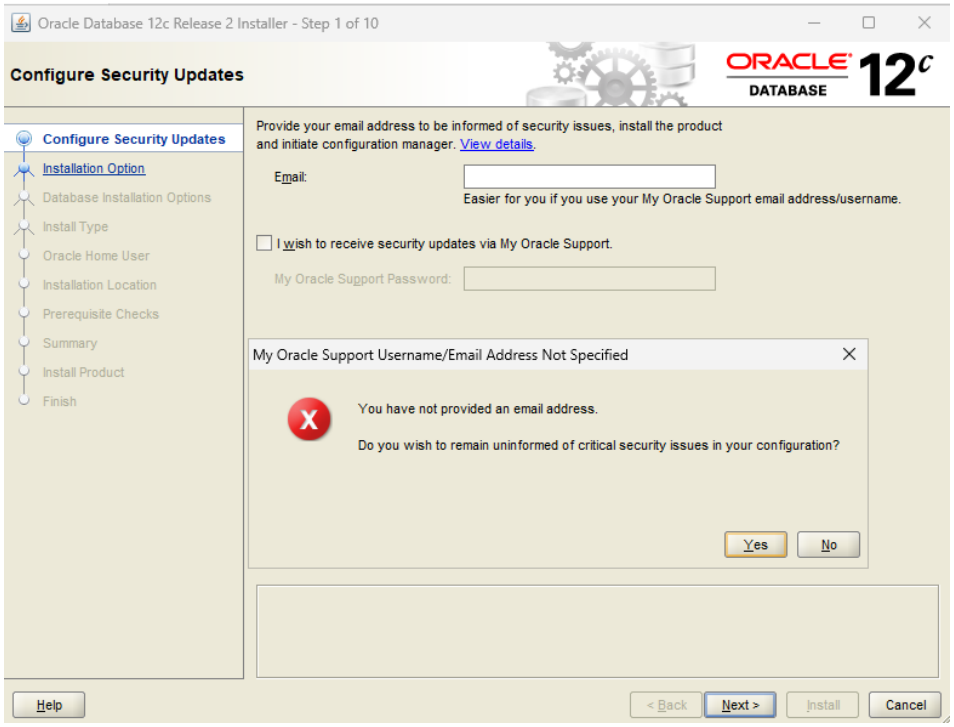
Инструкция описывает установку Oracle Database, создание базы данных для Global, экспорт схемы в дамп с помощью expdp и загрузку дампа с помощью `impdp`.

1.1 Установка Oracle Database

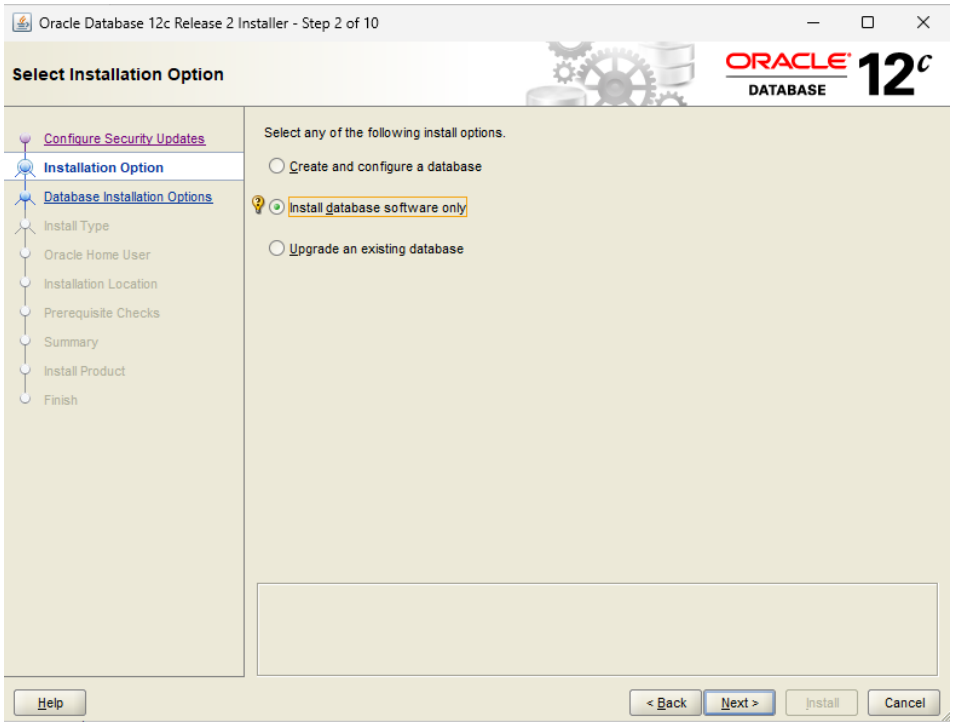
1.1.1 Windows

Перед созданием базы установите серверную часть Oracle Database.

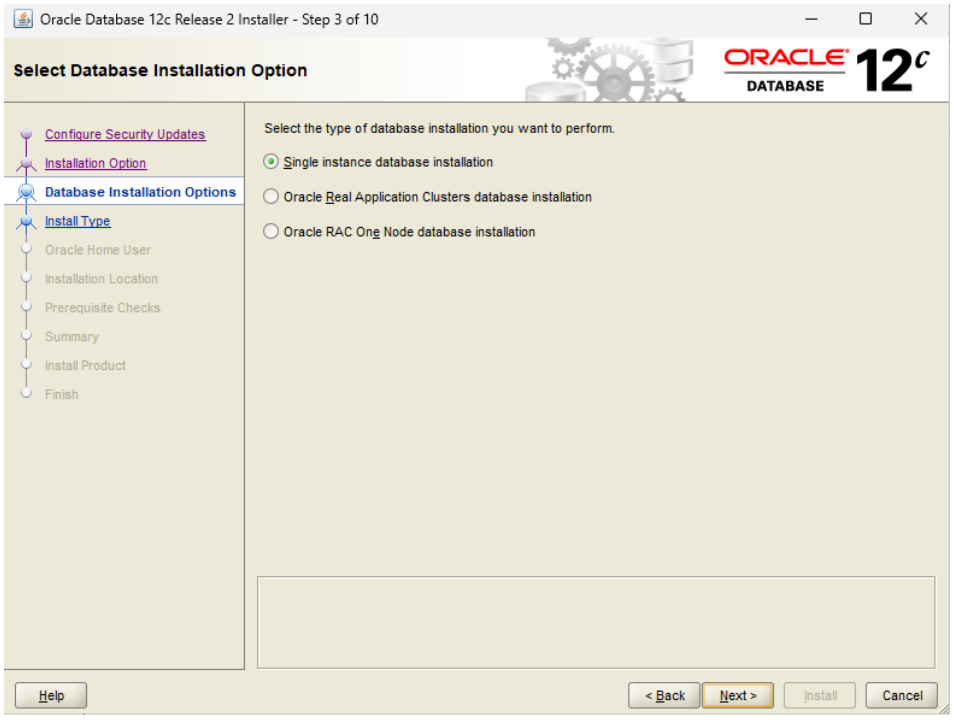
1. Запустите установщик Oracle Database.
2. На шаге **Configure Security Updates** укажите параметры получения уведомлений Oracle. Если уведомления не используются, оставьте адрес электронной почты пустым и подтвердите продолжение установки.



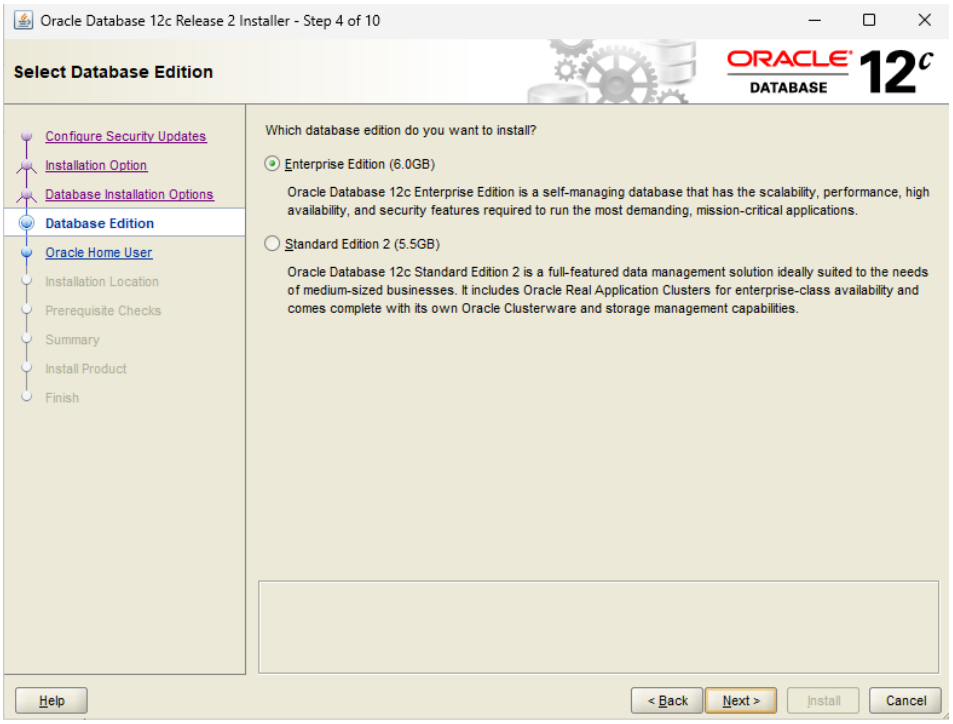
3. На шаге **Select Installation Option** выберите **Install database software only**. База данных будет создана отдельно после установки серверного ПО.



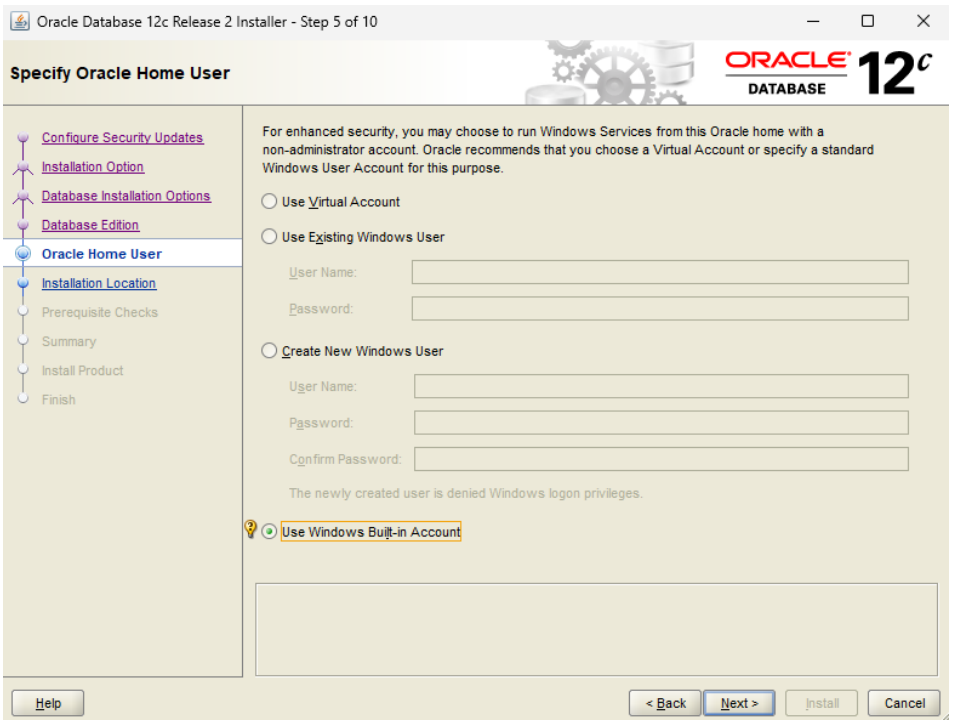
4. На шаге **Select Database Installation Option** выберите **Single instance database installation**.



5. На шаге **Select Database Edition** выберите редакцию Oracle Database, используемую в контуре. В приведенном примере используется **Enterprise Edition**.



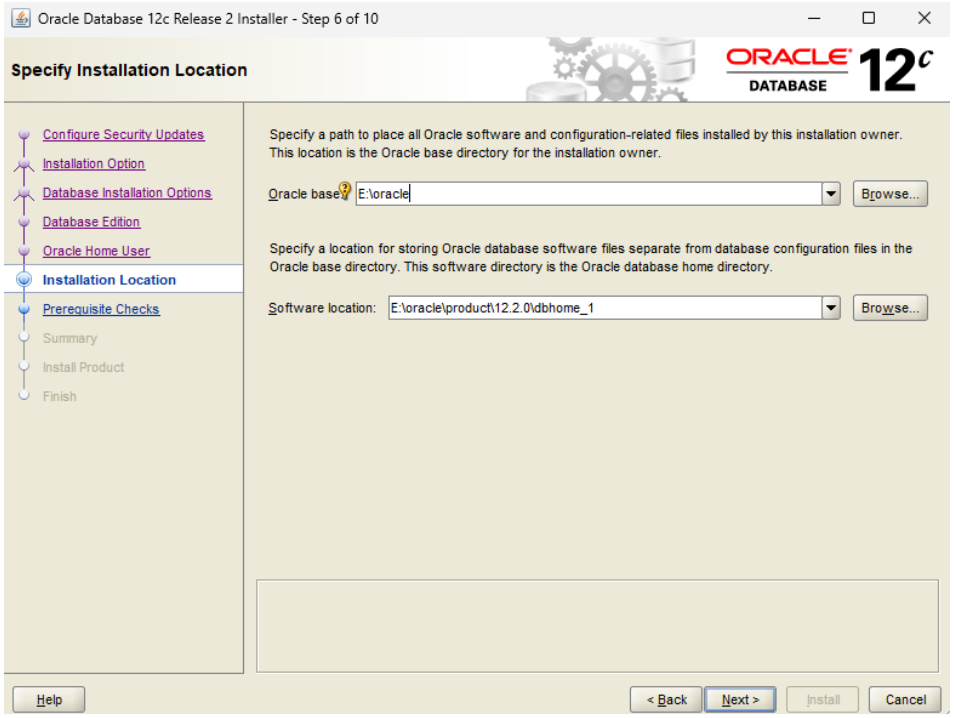
6. На шаге **Specify Oracle Home User** укажите учетную запись, от имени которой будут работать службы Oracle. В приведенном примере используется **Windows Built-in Account**.



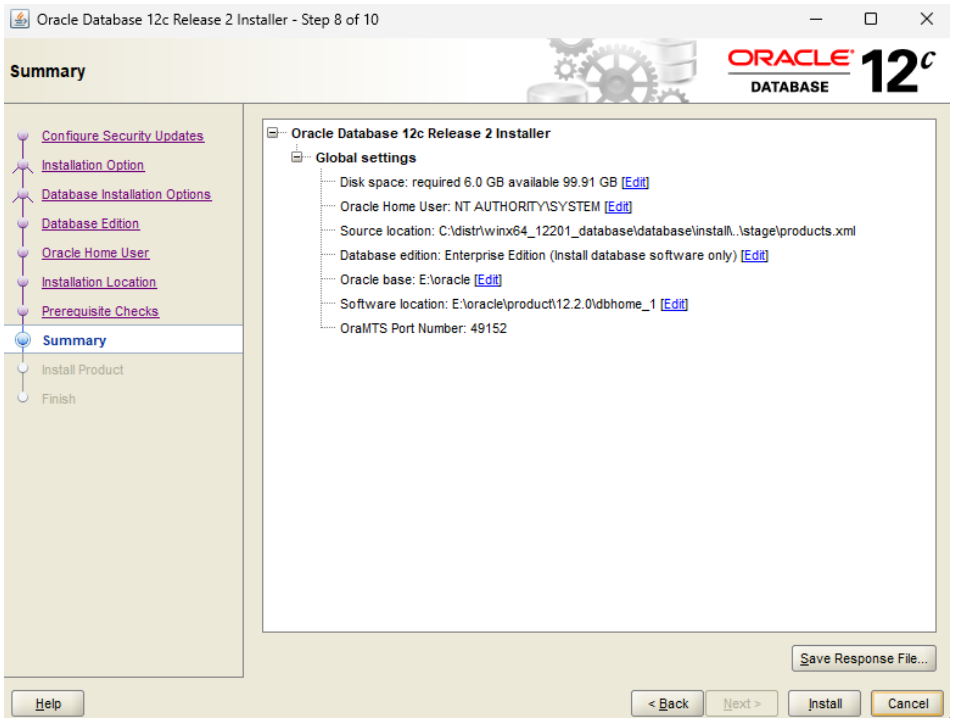
7. На шаге **Specify Installation Location** укажите:
   - **Oracle base** — корневой каталог Oracle;
   - **Software location** — каталог установки Oracle Database.

   Пример:

```
Oracle base: E:\oracle
Software location: E:\oracle\product\12.2.0\dbhome_1
```



8. Дождитесь завершения проверки предварительных требований.
9. На странице **Summary** проверьте выбранные параметры и нажмите **Install**.



После завершения установки настройте listener.

1.1.2 Linux

В Linux серверное ПО устанавливается из дистрибутива Oracle Database от имени пользователя Oracle.

1. Подготовьте каталоги `ORACLE_BASE` и `ORACLE_HOME` и предоставьте пользователю Oracle права на них.
2. Распакуйте дистрибутив Oracle Database.
3. Запустите установщик из каталога дистрибутива:

```
./runInstaller
```

4. В мастере установки выберите установку только серверного ПО и создание одиночного экземпляра базы данных.
5. Укажите `ORACLE_BASE` и `ORACLE_HOME`.
6. После проверки предварительных требований запустите установку.
7. Выполните системные скрипты, которые установщик предложит запустить от имени `root`.
8. После установки задайте окружение пользователя Oracle:

```
export ORACLE_BASE=<путь_к_oracle_base>
export ORACLE_HOME=<путь_к_oracle_home>
export PATH=$ORACLE_HOME/bin:$PATH
```

После установки настройте listener и создайте базу данных.

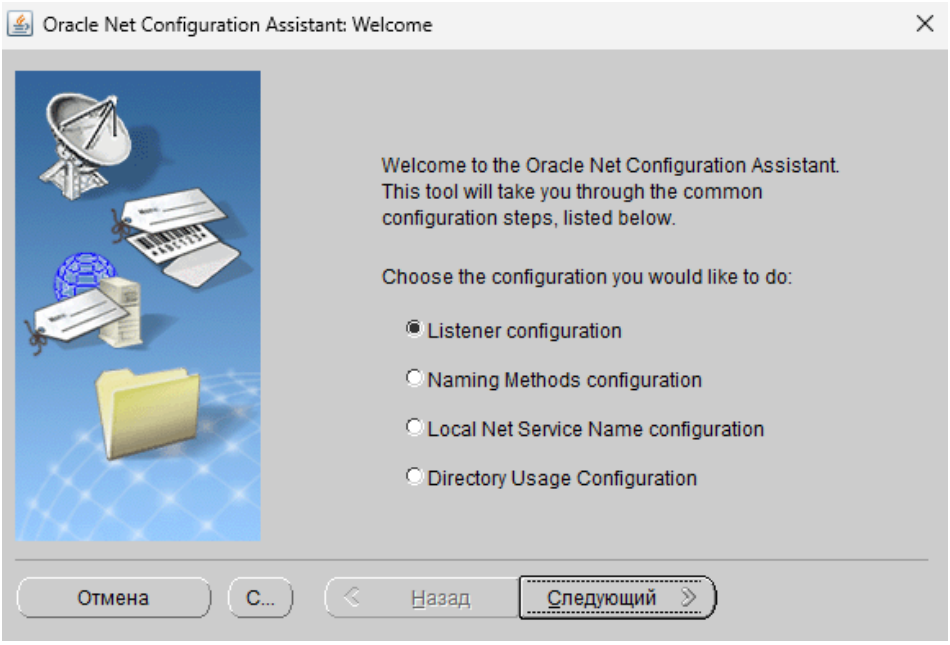
1.2 Настройка listener

Listener принимает сетевые подключения к Oracle Database. Для настройки используется Oracle Net Configuration Assistant.

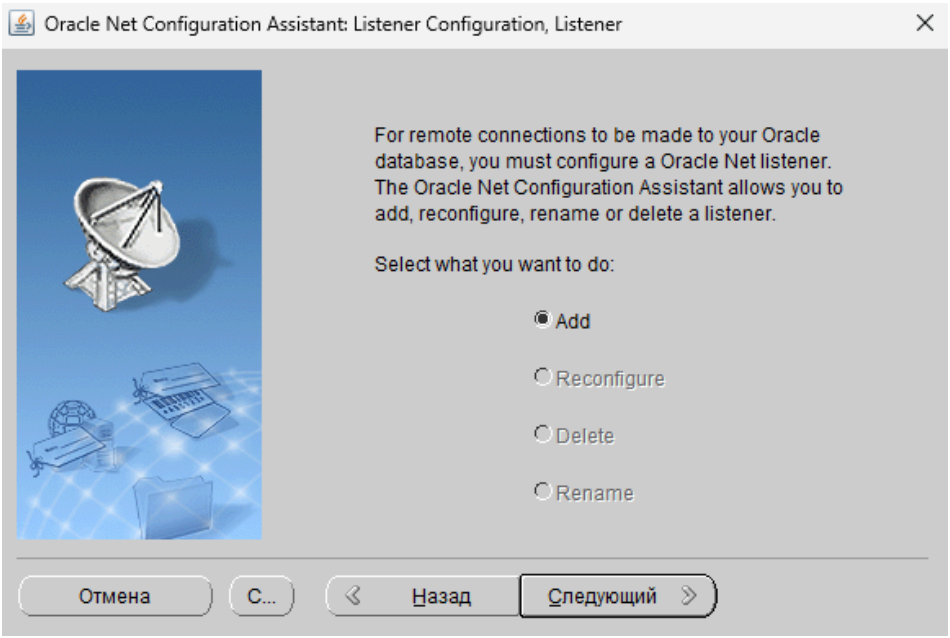
В Windows Oracle Net Configuration Assistant можно запустить из меню Oracle или командой `netca`. В Linux запустите:

```
$ORACLE_HOME/bin/netca
```

1. Выберите **Listener configuration**.



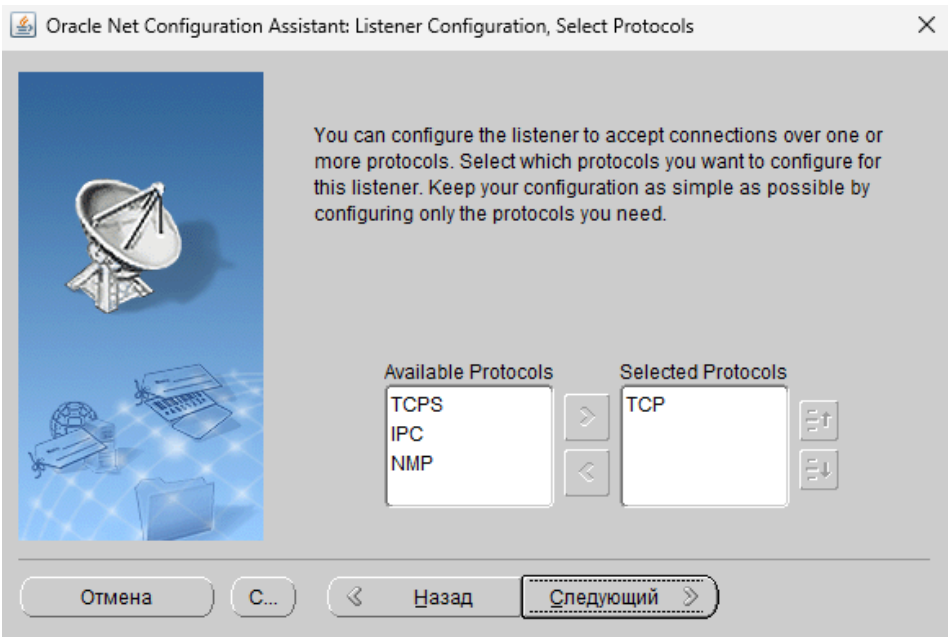
2. Выберите **Add** для создания listener.



3. Укажите имя listener. По умолчанию используется:

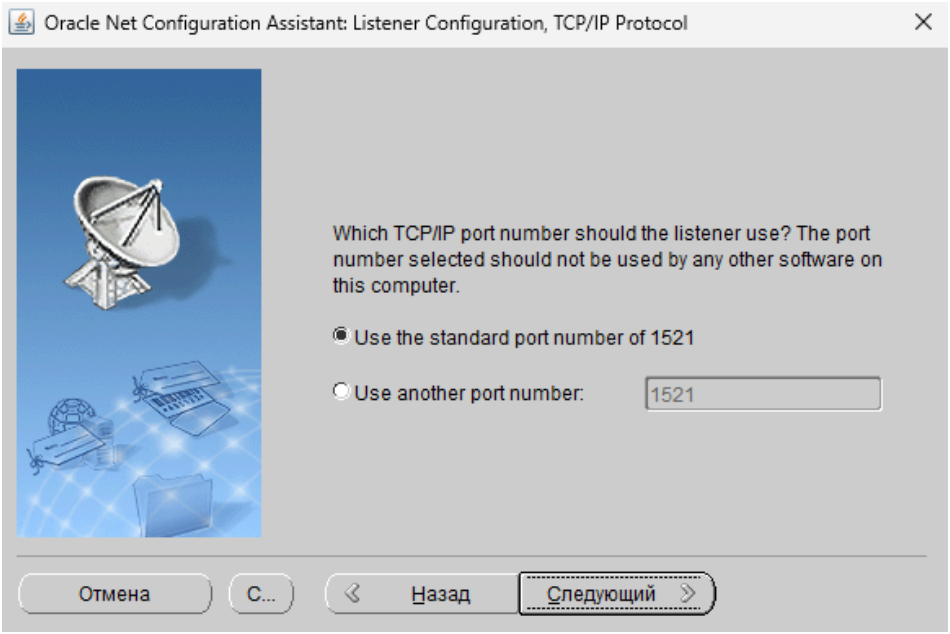
```
LISTENER
```

4. Выберите протокол **TCP**.



5. Укажите порт listener. Стандартный порт Oracle:

```
1521
```



6. Завершите настройку Oracle Net Configuration Assistant.

Проверьте состояние listener:

```
lsnrctl status
```

При необходимости запустите его:

```
lsnrctl start
```

1.3 Создание базы данных

База данных создается с помощью Oracle Database Configuration Assistant (DBCA).

В Windows запустите:

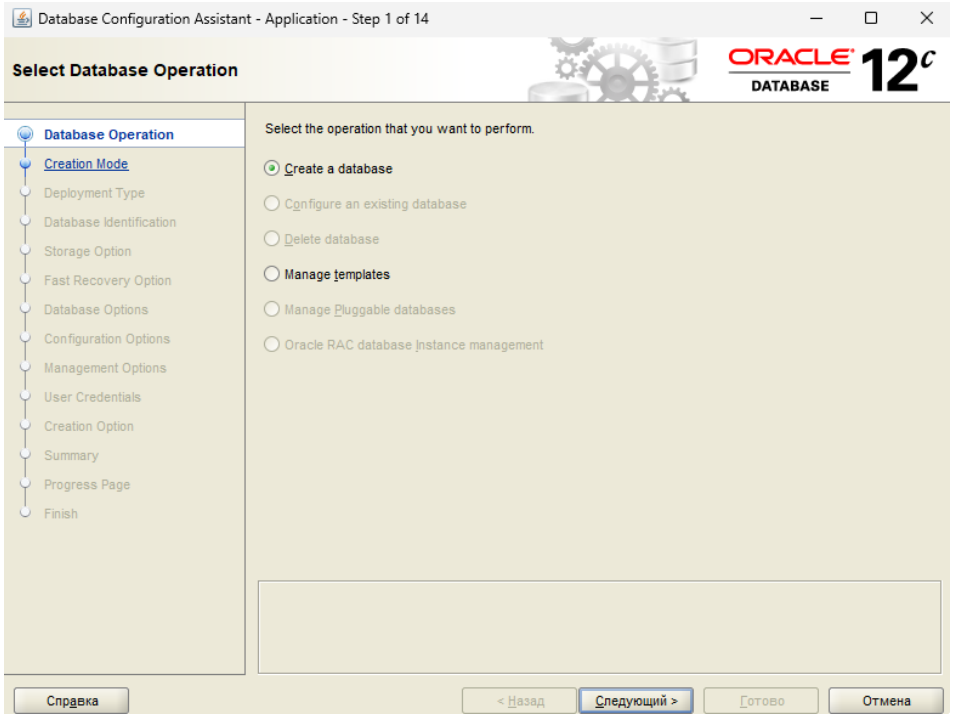
```
dbca
```

В Linux запустите:

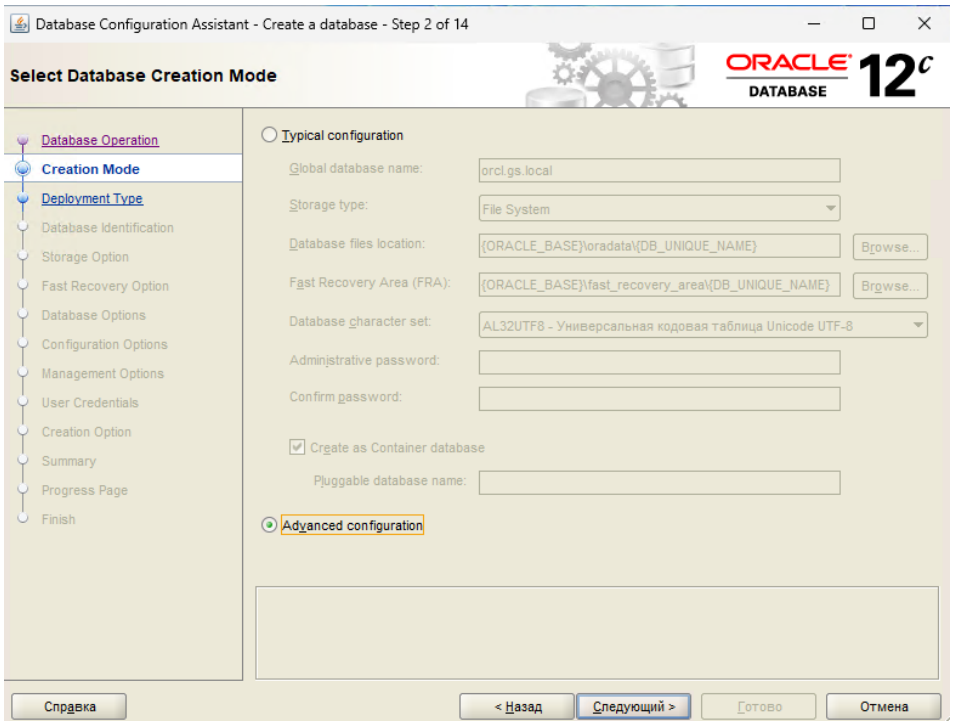
```
$ORACLE_HOME/bin/dbca
```

1.3.1 Основные параметры базы

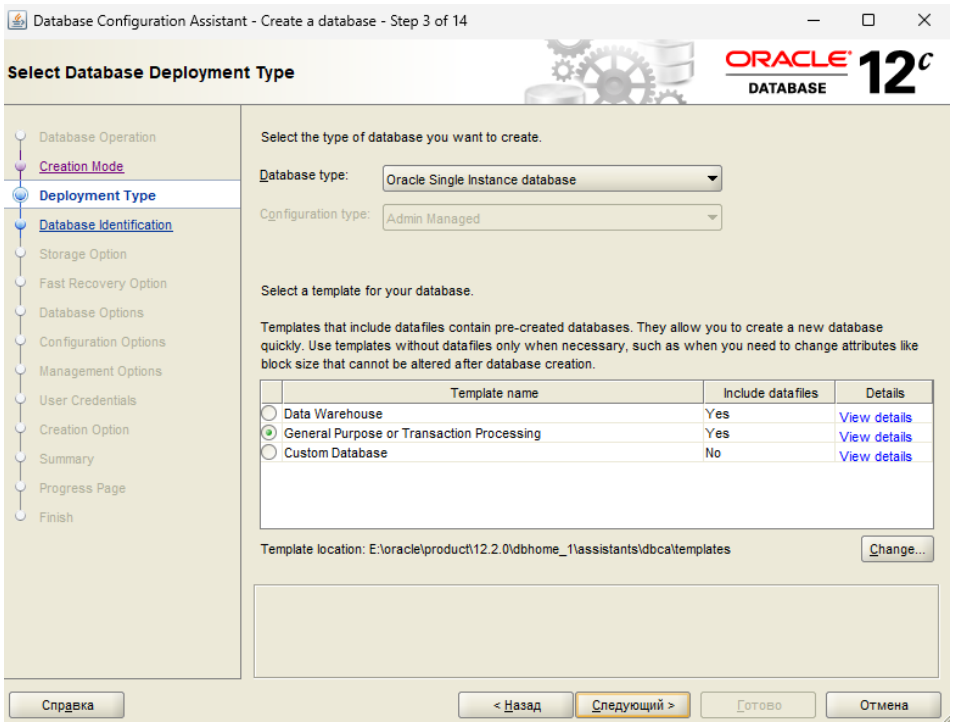
1. В DBCA выберите **Create a database**.



2. Выберите **Advanced configuration**. Расширенный режим позволяет вручную настроить память, кодировку, количество процессов и расположение файлов базы.

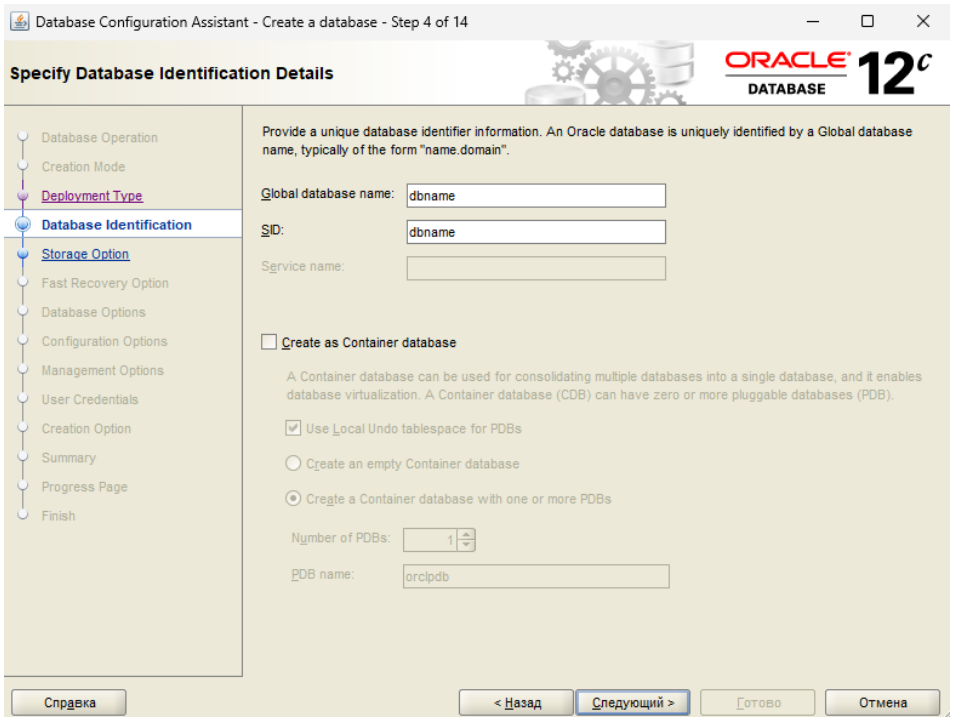


3. На шаге **Select Database Deployment Type** выберите:
   - **Oracle Single Instance database**;
   - шаблон **General Purpose or Transaction Processing**.



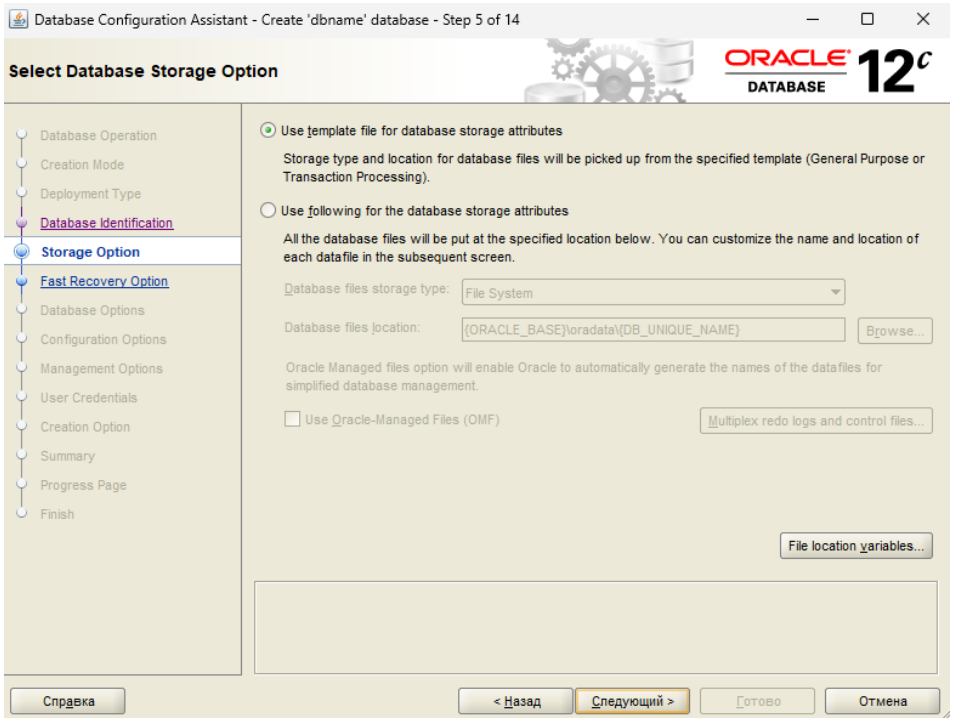
4. На шаге **Specify Database Identification Details** укажите:
   - **Global database name** — имя базы;
   - **SID** — идентификатор экземпляра.

   Для используемой конфигурации Oracle 12c флаг **Create as Container database** не устанавливается.

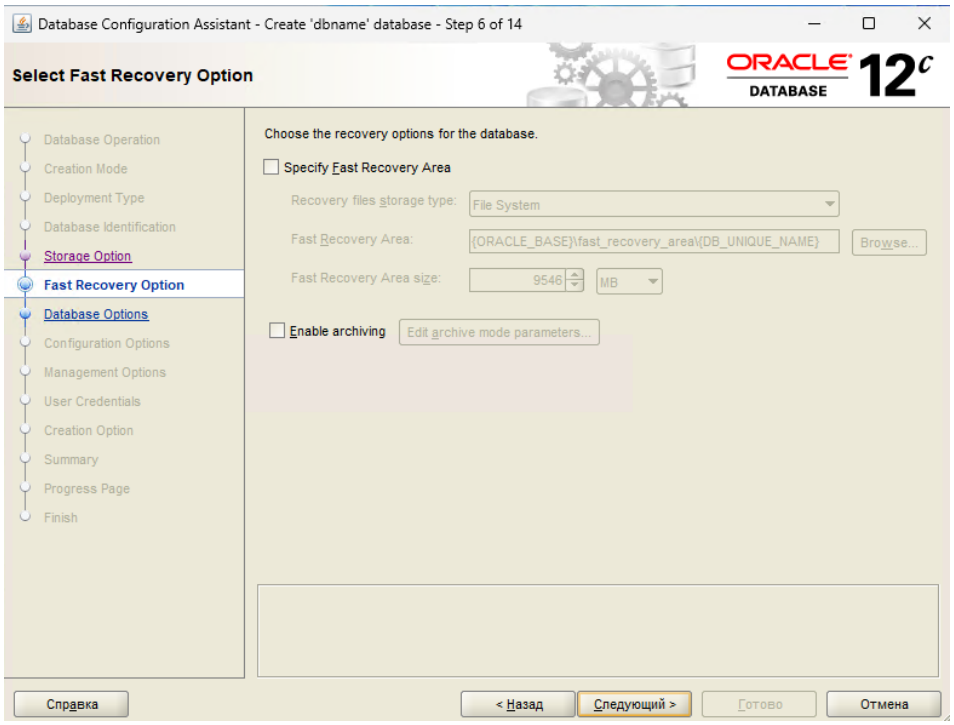


5. На шаге **Select Database Storage Option** укажите расположение файлов базы. При ис-

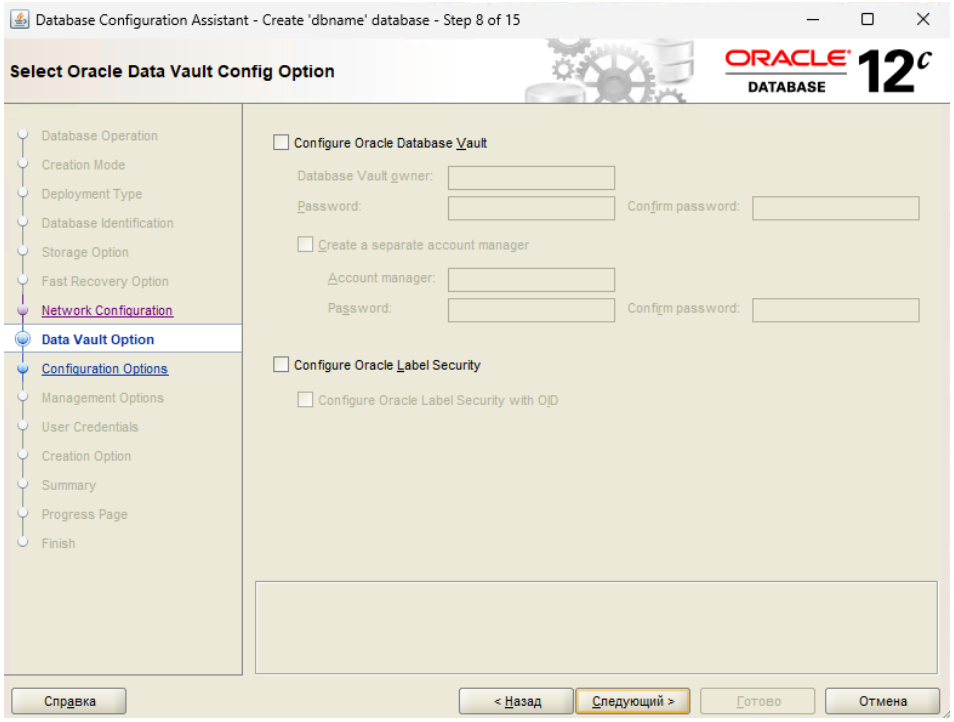
пользовании файловой системы можно оставить **Use template file for database storage attributes** и проверить расположение файлов перед непосредственным созданием базы.



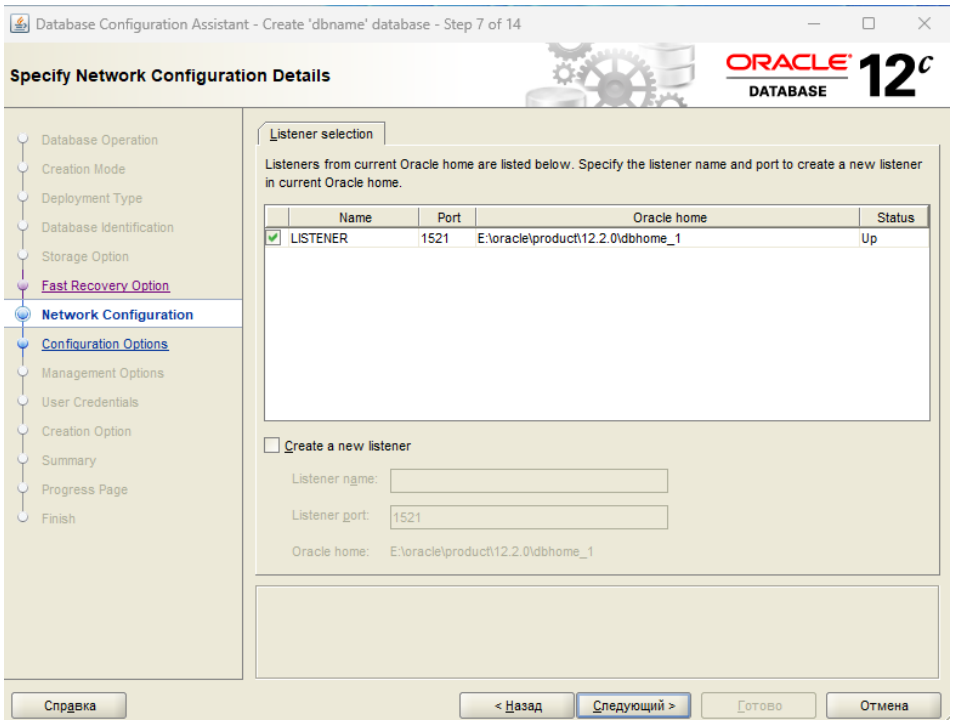
6. Настройте параметры восстановления и архивирования в соответствии с требованиями контура. В приведенном примере **Fast Recovery Area** и архивирование не используются.



7. На шаге настройки сети выберите ранее созданный `LISTENER` и проверьте используемый порт.



8. **Oracle Database Vault** и **Oracle Label Security** включаются только при необходимости. В приведенной конфигурации они не используются.



1.3.2 Настройка памяти

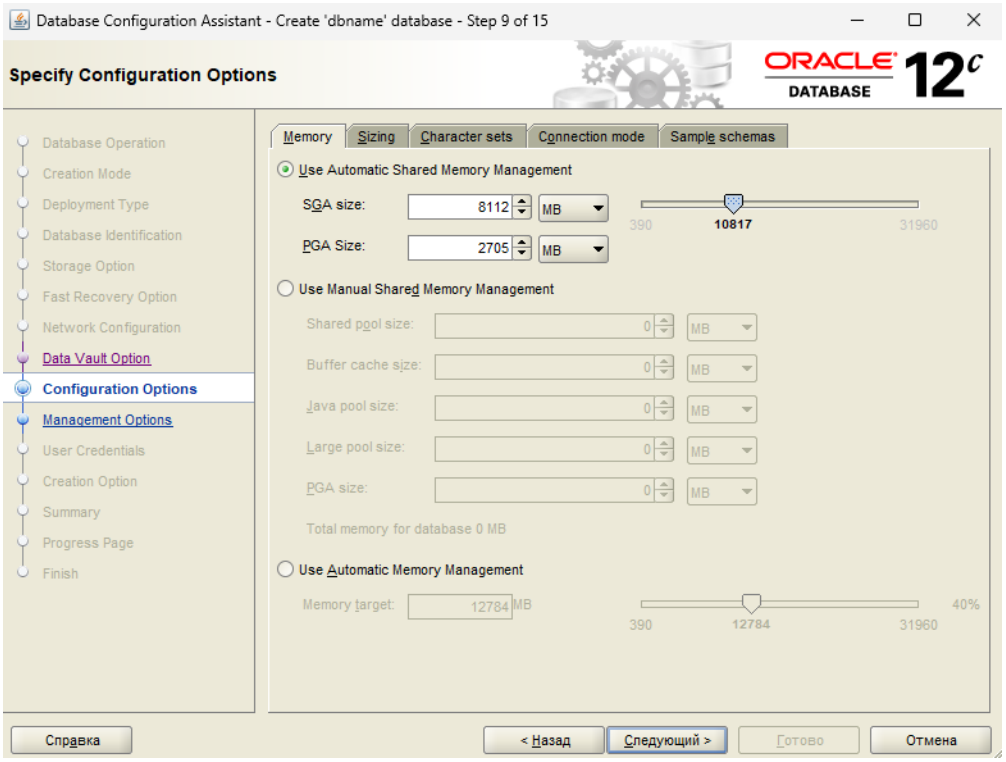
На странице **Configuration Options** откройте закладку **Memory** и используйте **Automatic Shared Memory Management**.

Оставляйте не менее 40% общей ОЗУ сервера для операционной системы и дискового кеша. Для Oracle используйте не более 60% ОЗУ.

Для тестовой базы с небольшим количеством пользователей и без ресурсоемких операций используйте начальную конфигурацию:

- SGA — 3 GB;
- PGA — 1 GB.

Для продуктивной базы значения выбираются с учетом объема данных, количества пользователей и характера нагрузки.



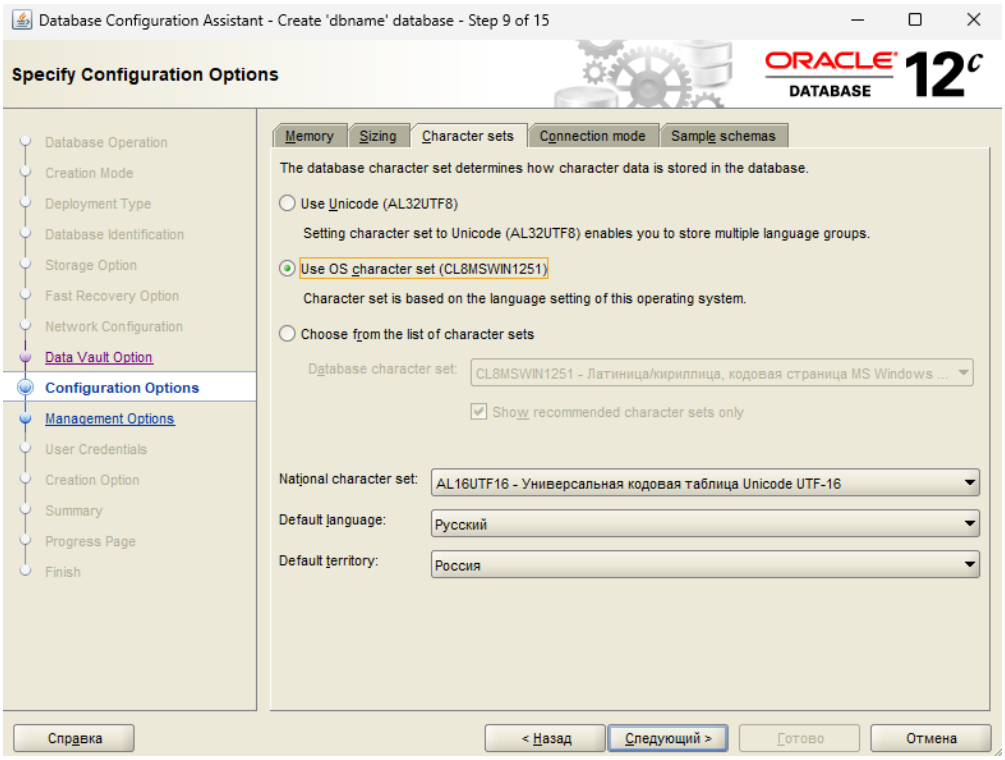
1.3.3 Настройка кодировки

На закладке **Character sets** выберите **Use OS character set (CL8MSWIN1251)**.

Для национального набора символов используется AL16UTF16.

Также укажите:

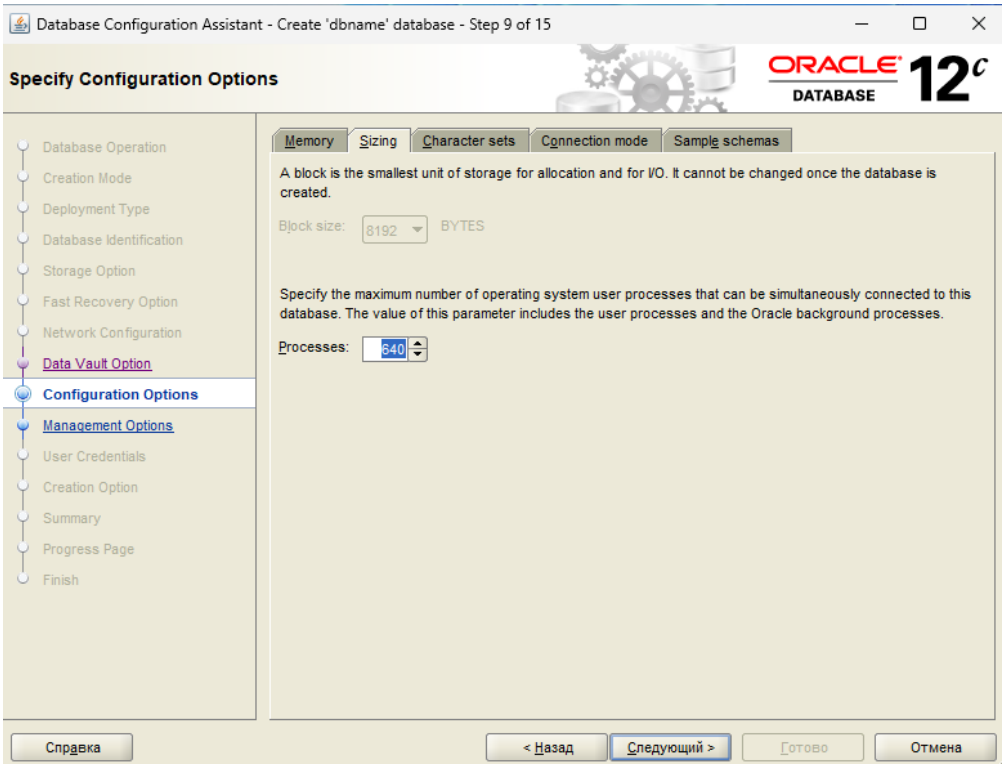
- **Default language** — Русский;
- **Default territory** — Россия.



1.3.4 Настройка количества процессов

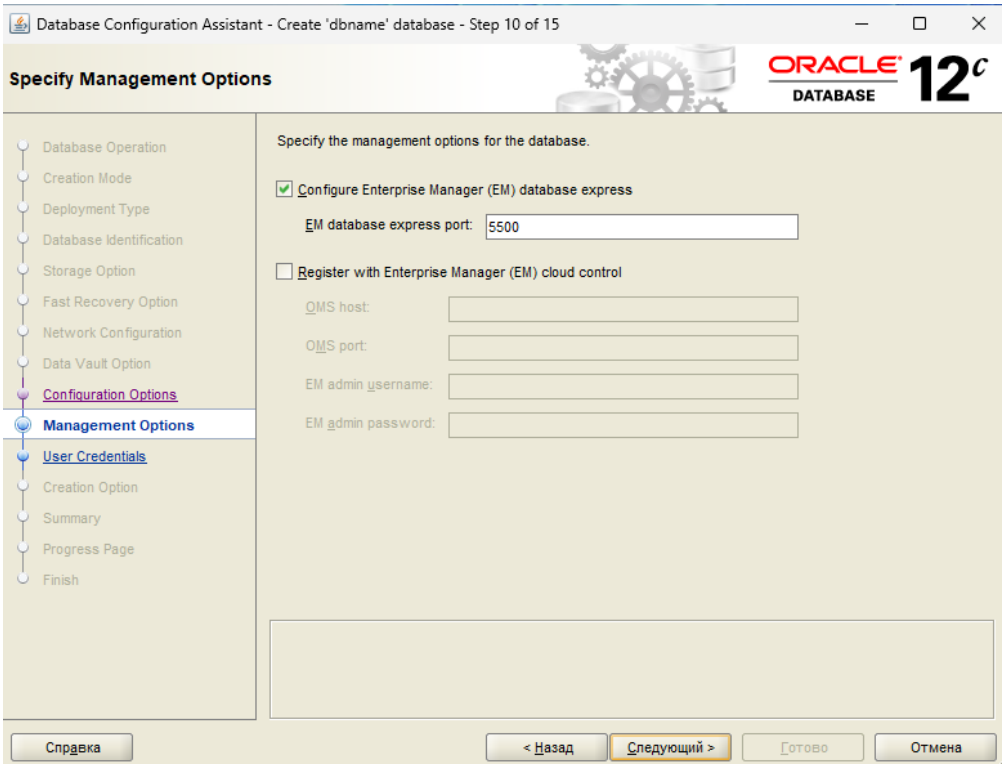
На закладке **Sizing** задайте значение **Processes**.

Для тестовой базы используйте значение в диапазоне от `320` до `1200`. Конкретное значение выбирается с учетом количества одновременных подключений и выполняемых процессов.



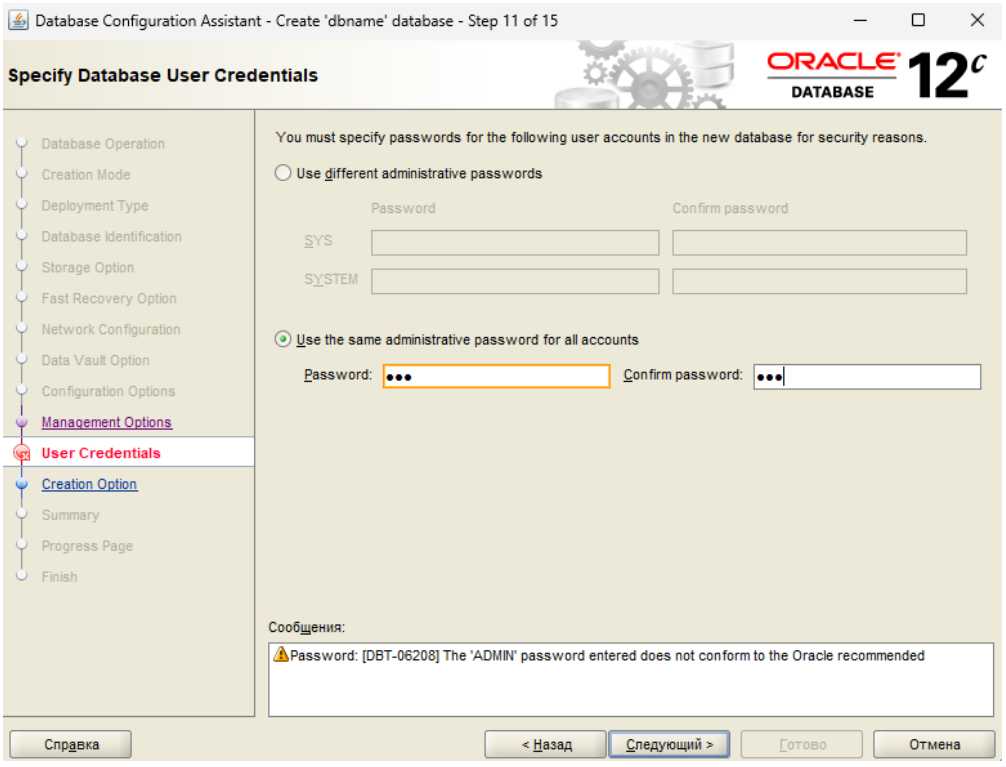
1.3.5 Настройка средств управления

На шаге **Management Options** настройте средства управления базой. В приведенном примере используется **Enterprise Manager Database Express** с портом `5500`.



1.3.6 Настройка административных пользователей

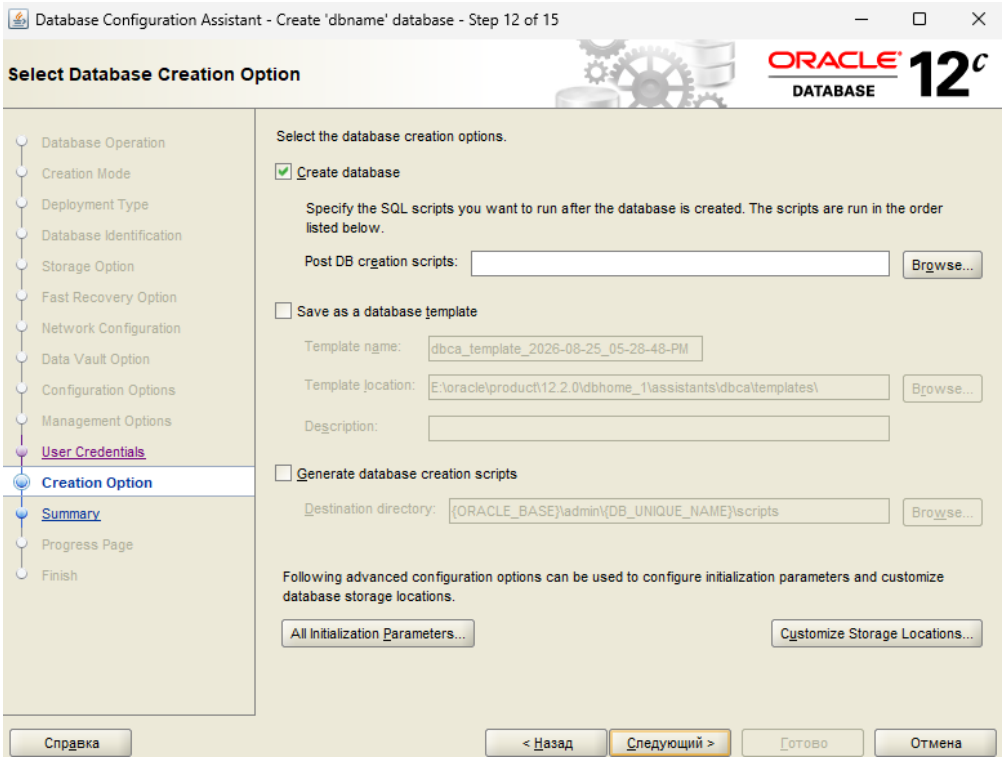
На шаге **User Credentials** задайте пароли административных пользователей Oracle.



1.3.7 Завершение создания базы

На шаге **Creation Options** установите **Create database**.

Перед созданием базы откройте **Customize Storage Locations** и проверьте состав и расположение файлов базы.



После проверки настроек запустите создание базы и дождитесь завершения DBCA.

Проверьте состояние экземпляра:

```
SELECT instance_name, status
FROM v$instance;
```

Проверьте состояние базы:

```
SELECT name, open_mode
FROM v$database;
```

1.4 Экспорт схемы с помощью expdp

Для создания дампа используется Oracle Data Pump Export (expdp).

1.4.1 Определение размера исходных данных

Перед экспортом определите объем исходных табличных пространств:

```
SELECT tablespace_name,
       ROUND(SUM(bytes) / 1024 / 1024 / 1024, 2) AS size_gb
FROM dba_data_files
GROUP BY tablespace_name
ORDER BY tablespace_name;
```

Полученные значения используйте при выборе размера датафайлов целевой базы.

1.4.2 Подготовка каталога Data Pump

Параметр DIRECTORY содержит имя объекта Oracle DIRECTORY, связанного с каталогом файловой системы на сервере Oracle.

Проверьте стандартный каталог DATA_PUMP_DIR:

```
SELECT directory_name, directory_path
FROM dba_directories
WHERE directory_name = 'DATA_PUMP_DIR';
```

Если используется отдельный каталог, создайте его на сервере Oracle и зарегистрируйте в базе:

```
CREATE OR REPLACE DIRECTORY GLOBAL_DP AS '<путь_к_каталогу>';
```

Предоставьте пользователю экспорта права на каталог:

```
GRANT READ, WRITE ON DIRECTORY GLOBAL_DP TO <user>;
```

1.4.3 Создание дампа

Выполните экспорт схемы:

```
expdp <user>@<service> DIRECTORY=DATA_PUMP_DIR DUMPFILE=<dump_file>.dmp LOGFILE=
↪<export_log>.log SCHEMAS=<schema>
```

Например, для схемы BTK:

```
expdp <user>@<service> DIRECTORY=DATA_PUMP_DIR DUMPFILE=btk.dmp LOGFILE=btk_export.
↪log SCHEMAS=BTK
```

После завершения экспорта проверьте:

- итоговое сообщение Data Pump;
- журнал экспорта;
- наличие файла .dmp в каталоге DATA_PUMP_DIR.

1.5 Импорт схемы с помощью impdp

Для загрузки дампа используется Oracle Data Pump Import (`impdp`).

Импорт схемы Global выполняется вместе со служебными SQL-скриптами, которые создают и подготавливают окружение базы.

1.5.1 Файлы импорта

Для импорта используются:

- *Import.bat* — пример командного файла для последовательного запуска импорта в Windows;
- *drop.sql* — подготовка схемы ВТК к повторной загрузке;
- *pre_script.sql* — создание схемы ВТК, назначение табличных пространств, квот и необходимых прав;
- *pre_post_script.sql* — подготовка объектов схемы после загрузки дампа;
- *pre_post_script2.sql* — дополнительная подготовка объектов после импорта;
- *post_script.sql* — завершающая компиляция и подготовка окружения Global;
- *dropAQ.sql* — вспомогательный скрипт удаления очередей Oracle AQ. В стандартной последовательности `Import.bat` автоматически не запускается.

1.5.2 Подготовка к импорту

Перед импортом убедитесь, что база данных создана и доступна.

Табличные пространства и датафайлы

У новой базы по умолчанию создается один основной датафайл для пользовательских данных. Перед импортом подготовьте табличные пространства и необходимое количество датафайлов.

Скрипт `pre_script.sql` назначает схеме ВТК квоты на табличные пространства:

- `USERSNEW`;
- `INDX`;
- `ACTNEW`.

Эти табличные пространства должны существовать до запуска импорта.

Датафайлы исходной базы

На исходной базе, из которой формируется дамп, получите список датафайлов:

```
SELECT tablespace_name, file_name, status, bytes
FROM dba_data_files;
```

По результату определите состав датафайлов исходной базы: табличные пространства, количество файлов, их расположение и размер.

Датафайлы целевой базы

На целевой базе, в которую будет загружен дамп, получите текущий список датафайлов:

```
SELECT tablespace_name, file_name, status, bytes
FROM dba_data_files;
```

Сопоставьте список с исходной базой и добавьте недостающие датафайлы в соответствующие табличные пространства.

Для добавления датафайла используйте `ALTER TABLESPACE`. Например:

```
ALTER TABLESPACE USERS
ADD DATAFILE 'D:\oradata\test\USERS02.DBF'
SIZE 100M
REUSE
AUTOEXTEND ON
NEXT 100M
MAXSIZE UNLIMITED;
```

В примере `D:\oradata\test\USERS02.DBF` — путь к создаваемому датафайлу. Для целевой базы укажите фактический каталог данных и нужное табличное пространство.

Если требуемого табличного пространства еще нет, создайте его до добавления дополнительных датафайлов.

Создайте табличное пространство `USERSNEW`:

```
CREATE TABLESPACE USERSNEW
DATAFILE '<путь_к_файлу>/usersnew01.dbf'
SIZE <начальный_размер>
AUTOEXTEND ON
NEXT <шаг_расширения>
MAXSIZE <максимальный_размер>;
```

Создайте табличное пространство `INDX`:

```
CREATE TABLESPACE INDX
DATAFILE '<путь_к_файлу>/indx01.dbf'
SIZE <начальный_размер>
AUTOEXTEND ON
NEXT <шаг_расширения>
MAXSIZE <максимальный_размер>;
```

Создайте табличное пространство `ACTNEW`:

```
CREATE TABLESPACE ACTNEW
DATAFILE '<путь_к_файлу>/actnew01.dbf'
SIZE <начальный_размер>
AUTOEXTEND ON
NEXT <шаг_расширения>
MAXSIZE <максимальный_размер>;
```

В Windows в <путь_к_файлу> укажите каталог данных Oracle. В Linux укажите каталог данных, используемый в установленном контуре.

Файлы и подключение

Перед запуском импорта:

1. Проверьте свободное место на дисках.
2. Скопируйте дамп `btk.dmp` в физический каталог, соответствующий `DATA_PUMP_DIR`.
3. Скопируйте SQL-скрипты и `Import.bat` в рабочий каталог.
4. Проверьте доступность сервиса Oracle:

   ```
   sqlplus <user>@<service>
   ```

5. В `Import.bat` замените значения примера на параметры целевой базы:
   - путь к `ORACLE_HOME`;
   - имя сервиса;
   - административные учетные данные;
   - имя дампа.

Предупреждение

`drop.sql` выполняет `DROP USER BTK CASCADE` и полностью удаляет существующую схему `BTK` вместе с принадлежащими ей объектами.

Запускайте импорт с удалением схемы только для базы, которую требуется полностью перезалить из дампа.

1.5.3 Порядок импорта

`Import.bat` выполняет операции в следующем порядке.

1. Запускает `drop.sql` под `SYSDBA`.
2. Запускает `pre_script.sql` под `SYSDBA`.

   Скрипт создает владельца схемы `BTK`, назначает табличные пространства и квоты, предоставляет необходимые права и создает вспомогательное окружение.
3. Запускает `impdp` и загружает дамп:

   ```
   impdp "<sys_user>/<password>@<service> as sysdba" directory=DATA_PUMP_DIR
   ↪dumpfile=btk.dmp logfile=btk.log
   ```

4. Запускает `pre_post_script.sql` от имени схемы BTK.
5. Запускает `pre_post_script2.sql` под SYSDBA.
6. Запускает `post_script.sql` под SYSDBA.

`post_script.sql` перекомпилирует объекты, перестраивает необходимые индексы, создает и подготавливает окружение GLOBAL_SYSTEM, синхронизирует административные роли и собирает статистику схемы BTK.

Для Linux используется та же последовательность запуска `sqlplus` и `impdp`, но команды выполняются из `$ORACLE_HOME/bin` или из окружения пользователя Oracle. Вместо `Import.bat` можно использовать shell-скрипт с теми же шагами.

1.5.4 Проверка результата

После импорта проверьте журналы:

- `drop.log`;
- `pre.log`;
- `btk.log`;
- `pre_post.log`;
- `pre_post2.log`;
- `post.log`.

Проверьте наличие невалидных объектов:

```
SELECT owner,
       object_type,
       object_name
FROM dba_objects
WHERE status = 'INVALID'
  AND owner IN ('BTK', 'GLOBAL_SYSTEM')
ORDER BY owner, object_type, object_name;
```

Проверьте, что необходимые табличные пространства доступны, а датафайлы созданы и доступны Oracle.

После успешного импорта проверьте подключение Global к базе данных и выполнение основных операций системы.

2 Установка и настройка сервера Global3

Инструкция описывает подготовку окружения, установку Oracle Client, развертывание сервера приложений GlobalServer, настройку подключения к Oracle и запуск сервиса `global3`.

2.1 Необходимое программное обеспечение

Для установки сервера приложений требуется:

- Java — требования к версии и используемой JDK/JRE приведены в разделе Требования к Java;
- `libaio1` — для Debian 12;
- `patchelf`, `libaio1t64` — для Debian 13;
- `tar`;
- `unzip`.

2.2 Установка Oracle Client 11

Требуется полная установка Oracle Client с типом **Administrator**.

В примерах ниже предполагается, что Oracle Client установлен в каталог:

```
/opt/oracle
```

Если используется другой путь, подставьте его в команды и переменные окружения.

2.2.1 Настройка файла tnsnames.ora

Создайте файл `tnsnames.ora`:

```
/opt/oracle/network/admin/tnsnames.ora
```

Пример содержимого файла:

```
<ALIAS> =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = <IP_HOST>)(PORT = <PORT>))
    )
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = <DB_NAME>)
    )
  )
```

Параметры файла:

- `<ALIAS>` — логический псевдоним базы, который будет использоваться приложением;
- `<DB_NAME>` — имя сервиса базы данных;
- `<IP_HOST>` — адрес сервера СУБД;
- `<PORT>` — порт подключения к СУБД.

2.2.2 Настройка прав доступа к Oracle Client

Создайте отдельную группу и пользователя, от имени которого будет работать сервер приложений:

```
sudo groupadd global
sudo useradd -g global -m -s /bin/bash global
```

Выдайте права на каталог Oracle Client:

```
sudo chown -R global:global /opt/oracle/
```

2.2.3 Настройка локали Oracle

При необходимости настройте локаль для корректного отображения русских символов:

```
export NLS_LANG=RUSSIAN_RUSSIA.CL8MSWIN1251
```

2.2.4 Проверка необходимых библиотек

Debian 12

Установите пакет `libaio1`. Он используется Oracle Database в Linux для асинхронного ввода-вывода:

```
sudo apt update
sudo apt-get install libaio1
```

Debian 13

Установите пакет `libaio1t64`:

```
sudo apt-get update
sudo apt-get install libaio1t64
sudo ln -s /usr/lib/x86_64-linux-gnu/libaio.so.1t64 /usr/lib/x86_64-linux-gnu/libaio.
↪so.1
```

Проверьте, что в каталоге Oracle Client присутствуют следующие файлы:

- `ojdbc6.jar`;
- `ucp.jar`;
- `orai18n.jar`.

Обычно они находятся в корневом каталоге Oracle Client или в его подкаталогах, например `jdbc/lib`.

Если файлов нет, скопируйте их из дистрибутива Oracle Client или установите недостающие компоненты.

2.3 Развертывание сервера приложений GlobalServer

2.3.1 Подготовка каталогов и дистрибутива

Скачайте архив дистрибутива. Файл `globalserver.zip` передается через контактное лицо.

Создайте временный каталог и загрузите в него архив:

```
sudo mkdir -p /tmp/global
```

Создайте каталог для установки сервера:

```
sudo mkdir -p /opt/global/globalserver
```

Распакуйте дистрибутив:

```
sudo unzip /tmp/global/globalserver.zip -d /opt/global/globalserver
```

Назначьте владельцем каталога пользователя `global`:

```
sudo chown -R global:global /opt/global/globalserver
```

2.3.2 Настройка прав на скрипты

Выдайте права на выполнение основных скриптов:

```
sudo chmod ug+x /opt/global/globalserver/start.sh
sudo chmod ug+x /opt/global/globalserver/stop.sh
sudo chmod ug+x /opt/global/globalserver/globalscheduler.sh
```

2.3.3 Настройка parameters.sh

Откройте актуальный файл `parameters.sh` и проверьте параметры окружения. При необходимости скорректируйте их в соответствии с фактическим расположением Oracle Client и его библиотек.

Внимание

Изменения конфигурации выполняйте только в файле `parameters.sh`. При изменении конфигурации в других файлах работоспособность не гарантируется.

Пример минимальной работоспособной конфигурации без дополнительных библиотек:

```
# Для запуска сервера приложений с подключением к БД Oracle

# Путь к Oracle Client
export ORACLE_HOME="/opt/oracle"

# Путь к каталогу с tnsnames.ora
export TNS_ADMIN="/opt/oracle/network/admin"
```

(продолжается на следующей странице)

(продолжение с предыдущей страницы)

```
# Пути к библиотекам Oracle Client
export LD_LIBRARY_PATH=$ORACLE_HOME:$ORACLE_HOME/:$LD_LIBRARY_PATH

# Java-библиотеки и JDBC-драйвер Oracle Client
export EXT_LIBS=$ORACLE_HOME/ojdbc6.jar:$ORACLE_HOME/ucp.jar:$ORACLE_HOME/orai18n.jar
export EXT_LIBS_CLASSPATH="$EXT_LIBS/*"
```

Убедитесь, что указанные пути соответствуют фактическому расположению Oracle Client на сервере.

2.3.4 Настройка сервиса global3

Для управления сервером приложений через `systemd` используется сервис `global3`.

Скопируйте шаблон сервиса и создайте файл `global3.service`:

```
sudo cp /opt/global/globalserver/admin/linux/global3.service.origin /lib/systemd/
↪system/global3.service
```

Обновите конфигурацию `systemd` и включите сервис:

```
sudo systemctl daemon-reload
sudo systemctl enable global3
```

2.3.5 Настройка подключения к базе данных

Откройте файл `global3.config.xml` в каталоге конфигурации сервера приложений внутри директории:

```
/opt/global/globalserver/application
```

В раздел `<databases>` добавьте описание подключения к базе данных Oracle:

```
<database alias="<alias>">
    <users>
        <user name="<user>" password="<password>"/>
    </users>
</database>
```

Параметры подключения:

- `<alias>` — логический псевдоним базы данных, который используется приложением;
- `<alias>` должен совпадать с именем подключения, указанным в `tnsnames.ora`;
- `<user>` и `<password>` — учетные данные схемы Oracle, под которой сервер приложений подключается к базе данных.

После изменения сохраните файл конфигурации.

2.3.6 Запуск сервера приложений

Запустите сервис:

```
sudo systemctl start global3
```

Проверьте его состояние:

```
sudo systemctl status global3
```

При успешном запуске сервис должен перейти в состояние `active (running)`.

2.3.7 Устранение ошибки с библиотеками Oracle

При входе в Global может возникнуть ошибка:

```
/opt/oracle/libocijdbc11.so: libnnz11.so: невозможно задействовать исполняемый стек,
↪как требует разделяемый объект: Отказано в доступе.
```

Причина связана с тем, что некоторые библиотеки Oracle помечены как требующие исполняемый стек, а политика безопасности системы запрещает это.

Debian 12

Установите `execstack`:

```
sudo apt update
sudo apt install execstack
```

Если на сервере нет доступа в интернет, выполните следующие действия на другой машине с той же версией системы и архитектурой:

1. Скачайте пакет `execstack`:

   ```
   sudo apt update
   sudo apt download execstack
   ```

2. Передайте `.deb`-файл на сервер.
3. Установите пакет:

   ```
   sudo dpkg -i execstack_*.deb
   ```

Снимите флаг исполняемого стека у библиотек Oracle:

```
sudo execstack -c /opt/oracle/libnnz11.so
sudo execstack -c /opt/oracle/libocijdbc11.so
```

Перезапустите сервис `global3`.

Debian 13

Установите `patchelf` и снимите флаг исполняемого стека:

```
sudo apt update
sudo apt install patchelf
patchelf --clear-execstack /opt/oracle/libnnz11.so
patchelf --clear-execstack /opt/oracle/libocijdbc11.so
```

2.3.8 Настройка логирования

Настройка логирования сервера приложений выполняется в отдельных конфигурационных файлах и описана в разделе Логирование сервера приложений.

Важно

Конфигурационные файлы поставляются с закомментированным содержимым. Проверьте файлы конфигурации, настройте необходимые параметры и раскомментируйте их.

2.3.9 Проверка установки

После завершения настройки выполните проверку.

1. Убедитесь, что сервис запущен:

   ```
   sudo systemctl status global3
   ```

2. Проверьте журналы сервера приложений и `systemd`. Убедитесь, что в них нет критических ошибок.

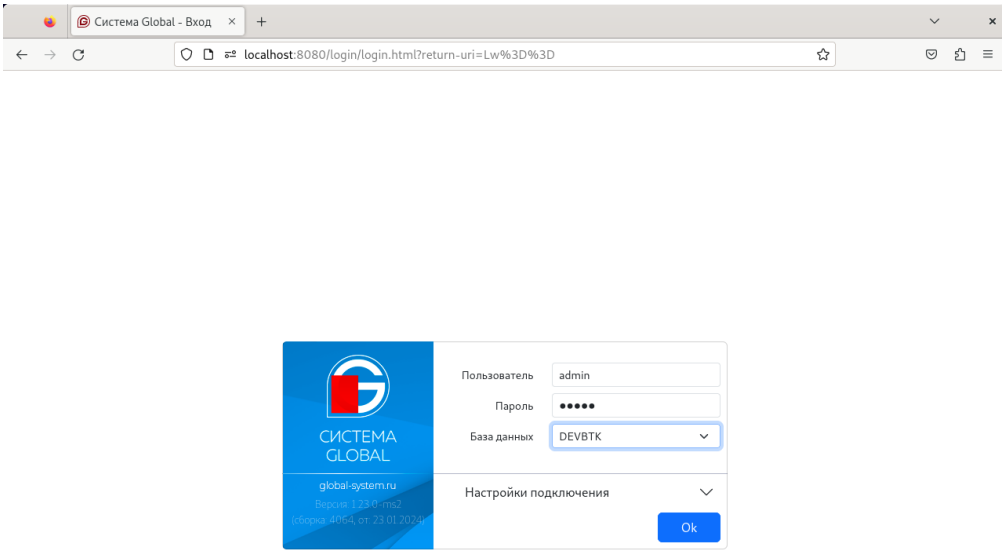
   Для проверки используйте:

   - логи `GlobalServer`;
   - журнал `systemd` для сервиса `global3`.

3. Проверьте подключение к базе данных Oracle:

   - выполните авторизацию в системе;
   - убедитесь, что базовые операции в приложении выполняются без ошибок.

Если проверки завершились успешно, сервер приложений GlobalServer установлен, подключен к Oracle и готов к работе.



3 Обновление сервера приложений

Инструкция описывает обновление установленного сервера приложений GlobalServer.

3.1 Подготовка окружения

Установите пакет `expect`:

```
sudo apt install -y expect
```

Создайте временный каталог для дистрибутива обновления:

```
sudo mkdir -p /tmp/global
```

Скопируйте в `/tmp/global` архив новой версии сервера приложений.

Обновите права на временный каталог и каталог сервера:

```
sudo chown -R global:global /tmp/global
sudo chown -R global:global /opt/global/globalserver
```

3.2 Подготовка скрипта config.sh

Скопируйте исходный конфигурационный скрипт обновления:

```
sudo cp /opt/global/globalserver/update/config.sh.origin /opt/global/globalserver/
↪update/config.sh
```

Откройте файл:

```
/opt/global/globalserver/update/config.sh
```

Укажите в нем:

- корректный alias базы данных;
- путь к временному каталогу, например `/tmp/global`;
- при необходимости другие параметры.

3.3 Настройка прав на скрипты обновления

Выдайте права на выполнение всем `.sh`-скриптам в каталоге `update`:

```
sudo find /opt/global/globalserver/update -type f -name '*.sh' -exec chmod a+x {} \;
```

Подробнее об утилите обновления смотрите в руководстве.

3.4 Запуск обновления

Запустите обновление сервера приложений:

```
/opt/global/globalserver/update/update.sh -m server
```

Дождитесь завершения скрипта и убедитесь по журналам и выводу команды, что обновление завершилось успешно.

3.5 Пример запуска обновления

```
root@oracle:~# /opt/global/globalserver/update/update.sh -m server

13.10.2026_17:03:08
####################
 * Global update *
 утилита обновления v 15.2
####################

План обновления:

Обновление сервера: Да.
Обновление jar:     Нет.
Установка релиза:   Нет.
----
Синхронизация прав доступа: Нет.
```

(продолжается на следующей странице)

(продолжение с предыдущей страницы)

```
Свободное место на диске:
104197 Мб
Поиск файлов дистрибутивов в каталоге /tmp/global
...
Найден дистрибутив сервера /tmp/global/globalserver.zip
Проверка архива ... No errors detected in compressed data of /tmp/global/globalserver.
↪zip.
Ok


####################
Обновление global...
Остановка менеджера заданий...
Failed to stop globalscheduler.service: Unit globalscheduler.service not loaded.
Поиск запущенного процесса менеджера заданий
Попытка 1
Менеджер заданий остановлен

Остановка сервера...
Поиск запущенного процесса сервера
Попытка 1
Сервер остановлен

распаковка /tmp/global/globalserver.zip
лог распаковки: /tmp/global/update_13.10.2026_170308/logs/unzip_server.log
версия установленного сервера: 1.27.3.9176
версия дистрибутива сервера  : 1.27.3.9176
обновление сервера не требуется
Запуск сервера...
проверяем доступность сервера по ssh...
Попытка 1
spawn ssh admin@localhost -p 2299
ssh: connect to host localhost port 2299: Connection refused
Сервер недоступен, пауза 10 сек...
Попытка 2
spawn ssh admin@localhost -p 2299
Password authentication
(admin@localhost) Password:
>
статус сервера: ЗАПУЩЕН
spawn ssh admin@localhost -p 2299 alter server mode normal
exit
Password authentication
(admin@localhost) Password:
OK
SSH Terminal session closed.

Запуск менеджера заданий...
Failed to start globalscheduler.service: Unit globalscheduler.service not found.

####################

Лог работы утилиты обновления: /tmp/global/update_13.10.2026_170308/logs/update_13.10.
↪2026_170308.log

Работа утилиты обновления завершена.
```

3.6 Проверка parameters.sh

После обновления сравните используемый файл `parameters.sh` с файлом, поставляемым в новой версии.

При необходимости перенесите актуальные параметры в настроенный `parameters.sh`.

После изменения конфигурации выполните:

```
sudo systemctl daemon-reload
sudo systemctl restart global3
```