



Установка «Ма3» на Astra Linux с СУБД Microsoft SQL

**РУКОВОДСТВО
АДМИНИСТРАТОРА**

СОДЕРЖАНИЕ

1. Общие сведения.....	1
Архитектура системы «Ma-3».....	1
Структура среды функционирования «Ma-3».....	2
2. Общее описание серверной конфигурации и состав пакета.....	4
3. Список пакетов.....	5
4. Распаковка компонентов.....	6
5. Настройка.....	7
Сервер лицензий.....	7
MSSQL.....	7
ODBC.....	9
dbaccess.....	11
appserver.....	15
Birt Report.....	16
Smartclient.....	18

1. Общие сведения

В данном разделе указаны сведения о технических и программных средствах, обеспечивающих работу сервера приложений «Ма-3».

Архитектура системы «Ма-3»

Общая архитектура системы «Ма-3» приведена на рисунке ниже:

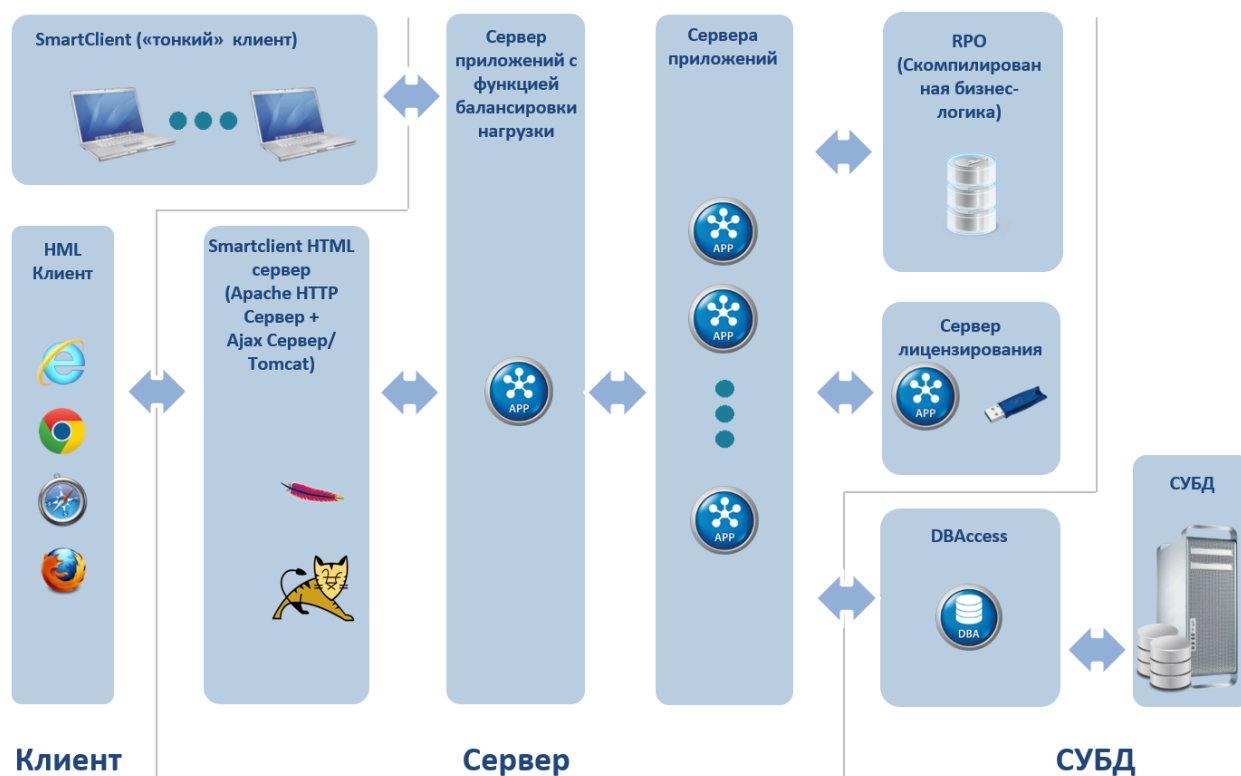


Рис. 1. Архитектура системы «Ма-3»

Клиент (Smart Client) – приложение, отвечающее за отображение пользовательского интерфейса (окна, графические элементы и т.д.) и обмен данными с сервером приложений. Запускается и работает на компьютере конечного пользователя. В системе клиентская часть реализована в виде отдельного приложения.

Также имеется специальный web-интерфейс, позволяющий выполнять различные операции в системе и не требующий установки какого-либо дополнительного ПО на стороне пользователя. Web-клиент представлен отдельным приложением выполняемым на стороне сервера, доступ к которому осуществляется пользователем с помощью стандартного web-браузера.

Сервер приложений (Application Server) – приложение, отвечающее за доступ к данным и их обработку. Данный компонент работает на специально выделенном сервере. Данный серверный компонент позволяет строить как малые, так и большие кластерные системы, что обеспечивает высокую производительность и дает возможность многопользовательской работы в системе. Количество одновременно работающих пользователей зависит от вычислительной мощности и примененной кластеризации.

«Ма-3» является ключевой частью этого сервера приложений, отвечающей за предотвращение несанкционированного доступа к информации, хранящейся и обрабатываемой в системе, регистрацию событий безопасности идентификации и аутентификации пользователей.

Сервер лицензирования – дополнительный компонент системы, отвечающий за соблюдение права использования системы в рамках лицензионного договора с конечным пользователем системы.

Репозиторий логики (RPO) – компонент системы (файл), содержащий скомпилированный набор алгоритмов обработки данных. Обработка данных происходит в соответствии с логикой алгоритмов системы, написанных на специальном языке программирования AdvPL. Наличие собственного внутреннего языка позволяет выполнять доработки поставленной функциональности в соответствии с потребностями конечного пользователя.

Приложение DBAccess – промежуточный компонент системы, отвечающий за унификацию доступа к СУБД.

СУБД – компонент системы, отвечающий за хранение данных. Систему возможно построить с использованием СУБД различных производителей. В базовом варианте предполагается использование СУБД MSSQL.

Структура среды функционирования «Ма-3»

Общая структура среды функционирования «Ма-3» приведена на рисунке ниже:



Рис. 2. Среда функционирования «Ма-3»

В среду функционирования «Ма-3» входит:

1. Операционная система сервера приложений. В общем случае – Linux-совместимая серверная ОС. В базовом варианте предполагается ОС AstraLinux CE (Орёл).
2. Java Runtime Environment (JRE) – среда исполнения приложений JAVA
3. Apache HTTP сервер – WEB-сервер для работы WEB-клиента
4. Драйвера СУБД MSSQL – ODBC DataBase Connector (UNIXODBC и ODBC-MSSQL). Устанавливаются на сервере, где установлен компонент DBAccess.

Данное программное обеспечение является стандартным и не входит в состав поставки сервера приложений «Ма-3». Особенности настройки и установки данного ПО применительно к «Ма-3» рассмотрены ниже.

WEB-клиент, компоненты «сервер лицензирования» и DBAccess, а также СУБД MSSQL относятся к среде функционирования «Ма-3», однако могут быть установлены на другой (других) физической (или виртуальной) машине.

2. Общее описание серверной конфигурации и состав пакета

Варианты расположения компонент:

1. на одном сервере
2. на нескольких серверах

В следующей инструкции по установке будет рассматриваться вариант установки “1”

В случае установки варианта “2” расположение компонент на серверах будет такое:

1 сервер: сервер лицензий

2 сервер: сервер базы данных, odbc, dbaccess

3 сервер: appserver и reportserver

Список архивов:

- Application Server архив bin.7z
- Ma-3 Data архив ma3_data.7z
- APO архив apo.7z

- ReportServer архив reportserver.7z
- LicenseServer архив licenseserver.7z
- DbAccess архив dbaccess.7z

MSSQL в составе архивов нет. Его нужно будет скачать с репозитория Astra Linux и установить с помощью менеджера пакетов apt.

Дополнительных утилит для MSSQL так же нет в составе архивов, их нужно будет скачать с репозитория Debian и установить с помощью менеджера пакетов apt.

3. Список пакетов

- Linux Astra version CE 2.12.13 (orel)
- msodbcsql 13.1.9.2-1 amd64 ODBC Driver for Microsoft(R) SQL Server(R)
- msodbcsql17 17.4.2.1-1 amd64 ODBC Driver for Microsoft(R) SQL Server(R)
- mssql-server 14.0.3238.1-19 amd64 Microsoft SQL Server Relational Database Engine
- libodbc1 2.3.7 amd64 ODBC library for Unix
- unixodbc 2.3.7 amd64 Basic ODBC tools
- unixodbc-dev 2.3.7 amd64 ODBC libraries for UNIX (development files)
- mssql-tools 17.4.1.1-1 amd64 Tools for Microsoft(R) SQL Server(R)
- unixodbc-dev 2.3.7 amd64 ODBC libraries for UNIX (development files)
- 'libc++1' 3.5-2 amd64 LLVM C++ Standard library
- libsss-nss-idmap0 1.16.3-1ubuntu2 amd64 SID based lookups library for SSSD
- hostname 3.18+b1 amd64 utility to set/show the host name or domain name
- libcurl3 7.52.1-5+deb9u9 amd64 easy-to-use client-side URL transfer library (GnuTLS flavour)
- gdb 7.12-6 amd64 GNU Debugger
- libunwind8 1.1-4.1 amd64 library to determine the call-chain of a program - runtime
- python 3.5.3-1+deb9u1 amd64 Interactive high-level object-oriented language (version 3.5)

4. Распаковка компонентов

Распаковку компонент нужно произвести согласно конфигурации среды:

1. Создайте каталог **/opt/ma3** на всех серверах, определенных под среду «Ma-3».
2. Создайте каталог **/opt/ma3/appserver**.
3. распакуйте архивы **bin.7z**, **ma3_data.7z**, **apo.7z** в каталог **/opt/ma3/appserver**.
4. Распакуйте архив **dbaccess.7z** в каталог **/opt/ma3/**
5. Распакуйте архив **licenseserver.7z** в каталог **/opt/ma3/**

После распаковки по варианту расположения компонент “1” каталоге **/opt/ma3/** должны быть:

```
appserver/  
reportserver/  
dbaccess/  
licenseserver/  
msodbcsql17.deb  
sqljdbc4.jar
```

5. Настройка

Сервер лицензий

После распаковки добавить службу:

```
mv /opt/ma3/licenseserver/ma3license.service /etc/systemd/system/
```

Службой можно управлять через **systemctl**.

MSSQL

Для настройки MSSQL следует выполнить следующие действия:

1. Добавить репозиторий **debian** для установки вспомогательных пакетов:

```
echo "deb http://deb.debian.org/debian stretch main" > /etc/apt/sources.list.d/debian.list
```

2. Добавить сертификат Microsoft и добавить репозитории Microsoft:

```
wget --no-check-certificate -q0- https://packages.microsoft.com/keys/microsoft.asc |
sudo apt-key add

wget --no-check-certificate -q0- https://packages.microsoft.com/config/ubuntu/16.04/
mssql-server-2017.list >
/etc/apt/sources.list.d/mssql-server-2017.list

wget --no-check-certificate -q0- https://packages.microsoft.com/config/ubuntu/16.04/
prod.list >
/etc/apt/sources.list.d/prod.list
```

3. Обновить списки пакетов и установить сертификаты репозитория:

```
apt-get update
apt-get install ca-certificates
```

4. Установить вспомогательные пакеты и определить переменные пути:

```
apt-get install -y python
apt-get install -y --allow-unauthenticated libodbc1 unixodbc unixodbc-dev msodbcsql
mssql-tools unixodbc-dev 'libc++1' libsss-nss-idmap0 hostname libcurl3 gdb libunwind8

echo 'export PATH="$PATH:/opt/mssql-tools/bin"' >> ~/.bash_profile

echo 'export PATH="$PATH:/opt/mssql-tools/bin"' >> ~/.bashrc

source ~/.bashrc
```

Не обязательно: можно отключить репозиторий **debian**: удалить папку **sources.list.d** или закомментировать строку с адресом в файле **debian.list**.

```
nano /etc/apt/sources.list.d/debian.list
```

```
apt-get update
```

5. Установить MSSQL:

```
apt-get install -y mssql-server
```

6. Запустить setup для настройки MSSQL сервера:

```
/opt/mssql/bin/mssql-conf setup
```

7. Проверить статус службы:

```
systemctl status mssql-server
```

Запустить, если не запущено.

8. Задать порт и ip для mssql-server:

```
/opt/mssql/bin/mssql-conf set network.ipaddress 127.0.0.1
```

```
/opt/mssql/bin/mssql-conf set network.tcpport 1433
```

9. Перезапустить службу mssql-server для применения сетевых настроек:

```
systemctl restart mssql-server
```

10. Убедиться, что запущено:

```
systemctl status mssql-server
```

11. Убедиться, что mssql слушает указанный порт 1433. В данном случае ip 127.0.0.1:

```
ss -tln
```

```
root@astra2:/home/yowa# ss -tln
State      Recv-Q Send-Q           Local Address:Port           Peer Address:Port
LISTEN    0      128          *:8080                        *:*
LISTEN    0        5            *:7890                        *:*
LISTEN    0      128            *:22                          *:*
LISTEN    0        5      127.0.0.1:631                 *:*
LISTEN    0      128    192.168.68.231:1433          *:*
LISTEN    0      128      127.0.0.1:2947              *:*
LISTEN    0      128            *:12100                       *:*
LISTEN    0      128            :::22                          :::*
LISTEN    0      100           :::8088                       :::*
LISTEN    0        1      :::ffff:127.0.0.1:8005      :::*
root@astra2:/home/yowa#
```

12. Настройки mssql-server из консоли (не обязательно):

```
/opt/mssql/bin/mssql-conf set [параметр]
```

Параметры тут:

```
/opt/mssql/bin/mssql-conf list
```

После изменения настроек нужно перезапустить сервис:

```
systemctl restart mssql-server
```

 **Пример:** Отключение отправки статистики в Microsoft:

```
/opt/mssql/bin/mssql-conf set telemetry.customerfeedback false
```

13. В системе должна присутствовать EN UTF-8 кодировка и должна быть установлена как основная, проверяем, если другая, то меняем, иначе будет ошибка при работе с БД:

```
terminate called after throwing an instance of 'std::runtime_error' what():  
locale::facet::_S_create_c_locale name not valid
```

Выполните команды:

```
locale-gen "en_US.UTF-8"  
dpkg-reconfigure locales
```

и выберите **en_US.UTF-8 UTF-8**.

При выборе нужной локали появится предупреждение, что нужно выбрать локаль по умолчанию. Выберите снова en_US.UTF-8 UTF-8 и нажмите **OK**.

ODBC

Для настройки ODBC следует выполнить следующие действия:

1. Выполнить команды:

```
systemctl stop mssql-server  
systemctl edit mssql-server
```

2. Добавить в открывшийся файл:

```
[Service]  
Environment="LD_LIBRARY_PATH=/opt/mssql/lib"
```

3. Сохранить и закрыть файл (**ctrl+x, y, Enter**).

4. Запустить mssql-server:

```
systemctl start mssql-server
```

5. Создать ссылки **libssl.so** и **libcrypto.so** до драйверов с версией 1.0.2 и посмотреть, есть ли они среди драйверов:

```
root@astra2:/home/yowa# ls -l /usr/lib/x86_64-linux-gnu | grep libssl
ls: cannot access '-l': No such file or directory
libssl3.so
libssl.a
libssl.so
libssl.so.1.0.0
libssl.so.1.0.2
libssl.so.1.1
root@astra2:/home/yowa#
```

```
ls -l /usr/lib/x86_64-linux-gnu | grep libcrypto
```

```
root@astra2:/home/yowa# ls -l /usr/lib/x86_64-linux-gnu | grep libcrypto
ls: cannot access '-l': No such file or directory
libcrypto.a
libcryptopp.so.6
libcrypto.so
libcrypto.so.1.0.0
libcrypto.so.1.0.2
libcrypto.so.1.1
libcrypto++.so.6
libcrypto++.so.6.0.0
root@astra2:/home/yowa#
```

В выводе будет видно, что есть разные драйверы с разными версиями. Нужно выбрать драйверы с версией 1.0.2 и создать на него ссылку в **/opt/mssql/lib/**.

6. Добавить ссылки:

```
ln -s /usr/lib/x86_64-linux-gnu/libssl.so.1.0.2 /opt/mssql/lib/libssl.so
ln -s /usr/lib/x86_64-linux-gnu/libcrypto.so.1.0.2 /opt/mssql/lib/libcrypto.so
```

7. Дописать в **/etc/odbc.ini**:

```
[MSODBC]
Driver=ODBC Driver 17 for SQL Server
/* имя драйвера должно соответствовать названию блока в odbcinst.ini.
В данном случае используется [ODBC Driver 17 for SQL Server] */
Server=127.0.0.1,1433
/* адрес сетевого интерфейса и порт драйвера для dbaccess т.к. dbaccess и mssql
находятся на одном хосте, и mssql слушает 127.0.0.1 на 1433 порту */
/* можно указать localhost,1433 */
/* 1433 порт по умолчанию, если порт задан другой, то соответственно указываем его */
/* тут указываем реквизиты подключения непосредственно к mssql */
UID=sa
/* пользователь для базы данных */
PWD=[пароль для пользователя mssql]
Database=ms-r8
/* может ругаться, если база не создана, лучше добавлять после развёртывания дампа */
```

8. После каждого изменения **odbc.ini** нужно его переопределить:

```
odbcinst -i -s -l -f /etc/odbc.ini
```

```
odbcinst -i -d -f /etc/odbcinst.ini
```

По умолчанию устанавливается 2 драйвера, 13 и 17 версии, на официальном ресурсе Microsoft указано, что это нормально.

Так же драйверов может быть установлено больше, сам odbc является интерфейсом базы данных для **dbaccess**, и в **>odbc.ini** указывается именно тот драйвер, который будет использоваться.

```
cat /etc/odbcinst.ini
```

Вывод должен быть следующим:

```
[ODBC Driver 13 for SQL Server]
Description=Microsoft ODBC Driver 13 for SQL Server
Driver=/opt/microsoft/msodbcsql/lib64/libmsodbcsql-13.1.so.9.2
UsageCount=1
[ODBC Driver 17 for SQL Server]
Description=Microsoft ODBC Driver 17 for SQL Server
Driver=/opt/microsoft/msodbcsql17/lib64/libmsodbcsql-17.4.so.1.1
UsageCount=1
```

9. Проверить доступность порта и ip командой:

```
ss -tln
```

10. Подключиться из консоли с реквизитами MSSQL сервера:

```
sqlcmd -S [ip] -U[user]
```

Примечание:

Сам mssql по умолчанию тут: /opt/mssql

конфиг файл mssql тут: /var/opt/mssql

директория odbc драйверов тут: /usr/lib/x84_64-linux-gnu/odbc

odbcinst тут: /etc/odbcinst.ini

odbc.ini тут: /etc/odbc.ini

dbaccess

Для настройки dbaccess следует выполнить следующие действия:

1. настроить config тут: /opt/ma3/dbaccess/dbaccess.ini, со следующим содержимым:

```
[General]
Port=7890
/* порт для appserver */
LicenseServer=[ip адрес сервера лицензий]
```

```

LicensePort=[port для подключения к серверу лицензий]
/* адрес и порт сервера лицензий */
ByYouProc=1
LicenseLimit=60
AuditLog=0
AuditAll=0
ODBC30=1
AuditServer=
AuditPort=7891
AuditLevel=1
AuditSqlExec=1
AuditProcExec=1
AuditThreadComment=1
AuditERPUserName=1
AuditSingleThread=0
[MSSQL]
user=sa
password=
TableSpace=
IndexSpace=
environments=MSODBC
[MSSQL/MSODBC]
user=sa
password=
/* логин и пароль для подключения к бд (интерфейсу odbc) */
TableSpace=
IndexSpace=
/* эти настройки лучше всего делать через dbmonitor,*/
/* который можно запустить только с графическим интерфейсом */

/* т.к. пароль для базы данных должен быть в зашифрованном виде */
/* пароль шифруется при нажатии кнопки "Сохранить" после настройки */
/* во вкладке типа бд логина и пароля */

/* при настройке указать имя [MSSQL/MSODBC], как в /etc/odbc.ini */
/* при тесте соединения алиас указать [MSSQL/MSODBC], как в /etc/odbc.ini */

```

2. Установить службу:

```
mv /opt/ma3/dbaccess/ma3dbaccess.service /etc/systemd/system/
```

3. Проверить статус службы командой:

```
systemctl status ma3dbaccess
```

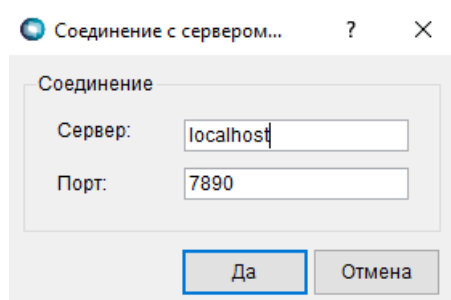
4. Запустить, если не запущено, командой:

```
systemctl start ma3dbaccess
```

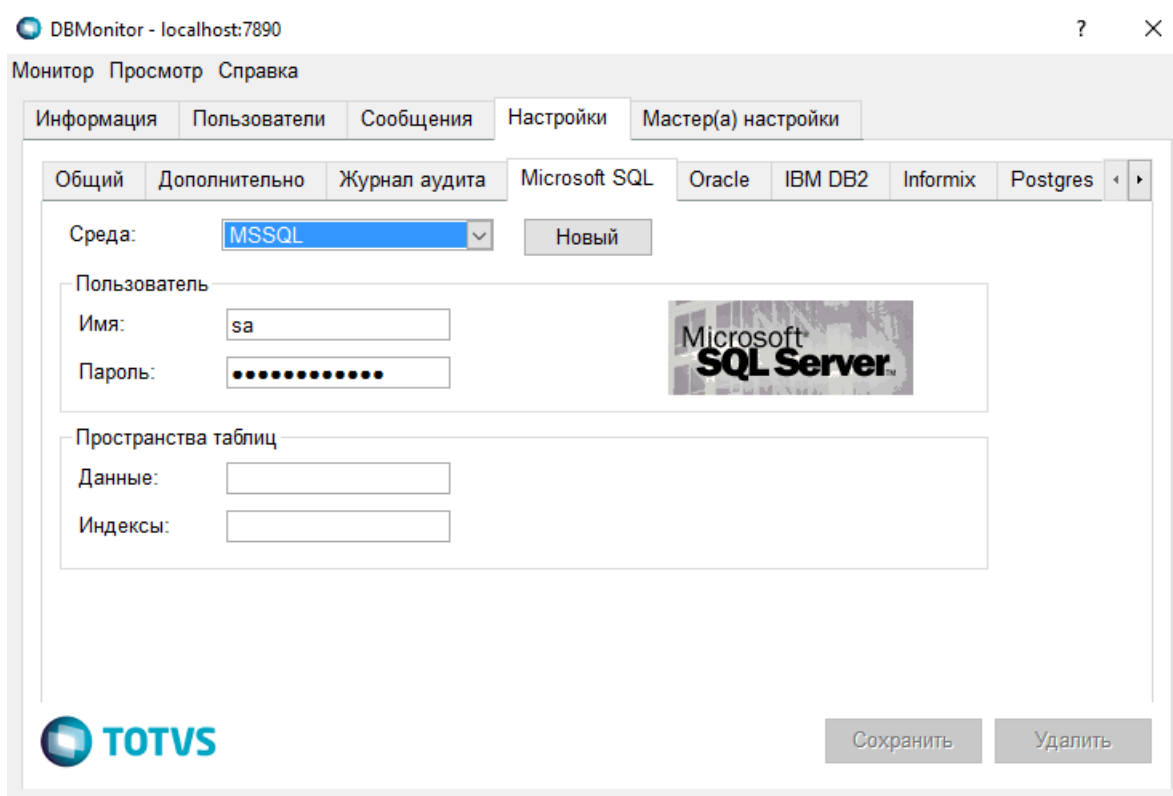
5. Запустить **dbmonitor**, который находится тут: **/opt/ma3/dbaccess/**:

⚠ Внимание: **dbmonitor** работает только с графическим интерфейсом, поэтому должен быть установлен на Astra Linux с графическим интерфейсом

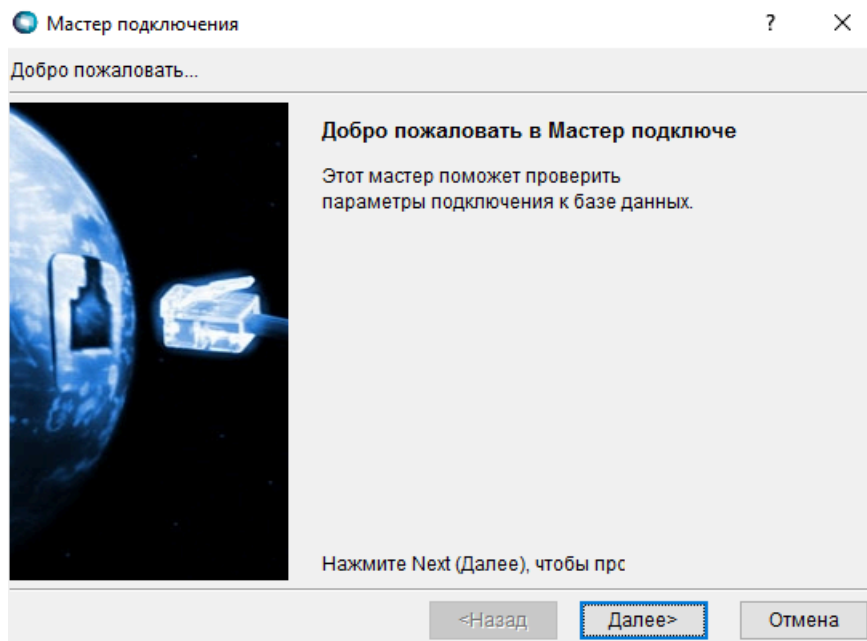
6. В окне соединения с сервером указать хост, на котором запущена служба ma3dbaccess (порт это параметр DBPORT в **/opt/ma3/dbaccess/dbaccess.ini**):



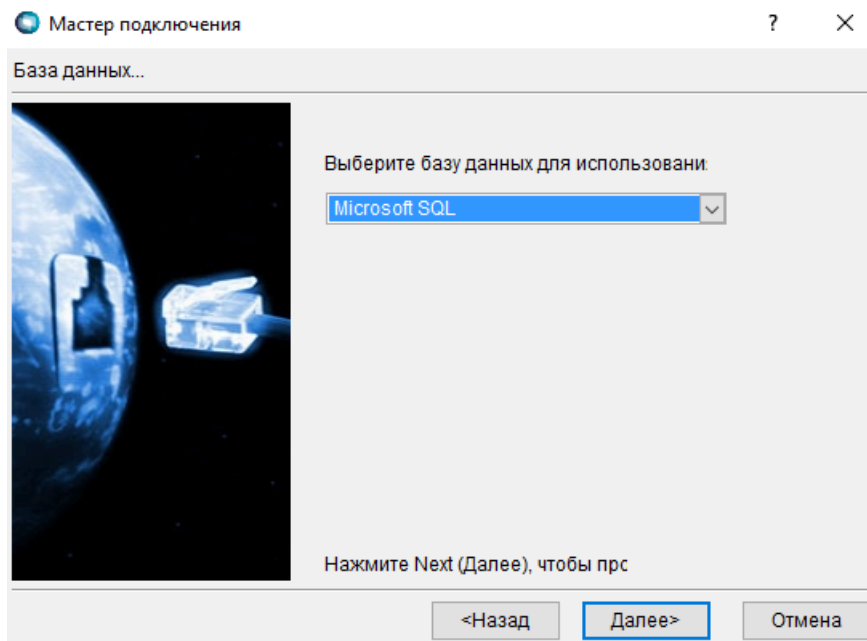
7. перейти во вкладку Настройки, выбрать вкладку Microsoft SQL ниже:



8. Назвать MSODBC как в настройке ODBC пункт 7 в `/etc/odbc.ini` "[MSODBC]":
Далее заполнить реквизиты доступа к базе данных – пользователь, пароль.
(пользователь должен иметь полные права на mssql-server) и нажать **Сохранить**.
9. Перейти во вкладку **Мастер настроек**, запустить проверку соединения в появившемся окне нажать кнопку **Далее**:



10. Выбрать базу данных Microsoft SQL:



Имя среды будет соответственно MSODBC.

Если тест пройден удачно, dbaccess настроен и его подключение к базе данных работает корректно.

appserver

Для настройки dbaccess следует выполнить следующие действия:

1. настроить config тут: /opt/ma3/appserver/appserver.ini, со следующим содержимым:

```
[ma3-appliance]
/* название среды указывается при подключении smartclient */
SOURCEPATH=/opt/ma3/appserver/apo/
/* путь, где лежит .rpo */
ROOTPATH=/opt/ma3/appserver/ma3_data/
STARTPATH=/system/
RPODB=TOP
RPOVERSION=120
LOCALFILES=CTREE
TRACE=0
LOCALDBEXTENSION=.dtc
PictFormat=RUSSIAN
DateFormat=RUSSIAN
DBSERVER=
/* ip адрес dbaccess */
/* написать как тут если сервер dbaccess и appserver устанавливаются на один сервер */
DBPORT=7890
/* порт dbaccess */
DBDATABASE=MSSQL
/* тип базы данных */
DBALIAS=MSSQL
/* алиас из /etc/odbc.ini вначале, например [MSODBC] */
/* он же в dbaccess.ini название секции, которая создавалась dbmonitor например [MSSQL/MSODBC] */
RPOLANGUAGE=English
REGIONALLANGUAGE=RUS
THEME=MA3
CFG_ENV_RPOLANGUAGE="r"
STRINGCODEPAGE=1
ENABLENUMBER=1
StartSYSInDB=1
TOPMEMOMEGA=1
ENABLENUMBER=1
RUTC_MODE=development
[General]
CONSOLEMAXSIZE=50919191
INSTALLPATH=/opt/ma3/appserver/bin/
CONSOLELOG=1
ConsoleFile=/opt/ma3/appserver/bin/console.log
DEBUGTHREADUSEDMEMORY=1
MAXSTRINGSIZE=500
ServerMemoryInfo=1
[webapp]
interface=np
port=8080
/* порт для web */
[LockServer]
Enable=1
Server=
/* ip адрес по котором сервер будет принимать подключения smartclient */
Port=12100
```

```
/* порт для smartclient */
[Drivers]
ACTIVE=TCP
[TCP]
TYPE=TCPIP
PORT=12100
[LICENSECLIENT]
server=
/* ip адрес сервера лицензий */
port=5555
/* порт сервера лицензий */
[TDS]
AllowApplyPatch=*
ALLOWEDIT=*
ENABLEDISCONNECTUSER=*
ENABLESENDMESSAGE=*
ENABLEBLOCKNEWCONNECTION=*
ENABLESTOPSERVER=*
```

2. Установить службу:

```
mv /opt/ma3/appserver/ma3appserver.service /etc/systemd/system/
```

Службой можно управлять через **systemctl**.

Birt Report

Birt Report воспользуется для работы встроенную jre 1.7bc Устанавливать jre или jdk в ОС не нужно - будет путаница из-за того, что веб приложение деплоится встроенной jre 1.7, а запускаться будет 1.8 или старше. По умолчанию устанавливается в /usr/local/totvswebserver Каталог WebViewer я переименовал в webviewer Каталог TOT VS...чётотам... большими...буквами... переименовал в totvswebserver.

1. Если у вас имя среды в **addpserver.ini ma3-appliance**, то создайте файл с таким именем, если имя среды другое, то укажите его вместо этого в части названия файла:

```
nano /opt/ma3/reportserver/server/webapps/webviewer/WEB-INF/ma3-appliance.properties
```

2. Написать:

```
username=[mssql_user]
password=[mssql_password]
```

Сохранить, закрыть.

3. Выполнить:

```
nano /opt/ma3/reportserver/server/webapps/webviewer/WEB-INF/config.properties
```

Внутри должно быть только:

```
username=[mssql_user]
password=[mssql_password]
```

Сохранить, закрыть.

4. Посмотреть порт на котором запускается tomcat. Пример:

```
<Service name="Catalina">  
<Connector port="8088" protocol="HTTP/1.1"
```

Не сохранять, закрыть.

5. положить sqljdbc4.jar сюда: **/opt/ma3/reportserver/server/webapps/webviewer/drivers.**

Я положу его к архиву, но его можно найти в интернете и скачать, имя должно быть именно такое: sqljdbc4.jar, т.е. если будет sqljdbc4.0.2.jar к примеру, то переименовать.

Если нет каталога drivers - создать.

6. Установить службу:

```
mv /opt/ma3/reportserver/ma3reportserver.service /etc/systemd/system/
```

Проверить доступность можно по адресам (8.8.8.8 это пример, вместо него поставить локальный ip и порт сервера, на котором работает birt)

```
http://8.8.8.8:1111  
http://8.8.8.8:1111/webviewer##
```

Если на второй странице вывести отчёт, то это будет проверка только механизма отчётов, т.е. birt не полезет в базу и не будет использовать шаблоны ма3.

7. Выполнить команду:

```
nano /opt/ma3/appserver/bin/appserver.ini
```

и добавить блок **reportservice**:

```
[REPORTSERVICE]  
ENABLED=1  
REPORTENGINE=TOMCAT  
WEBVIEWER=http://8.8.8.8:1111/webviewer/  
/* протокол, ip, порт и имя каталога webviewer бирта */  
/* порт, который в server.xml */  
/* ПОРТ НЕ ДОЛЖЕН СОВПАДАТЬ С ПОРТОМ appserver, указанным в appserver.ini в блоке  
[webapp] */  
/* иначе кто-то из них не запустится (appserver, birt) */  
REPORTDATABASE=MSSQL  
/* тип базы данных для birt чтоб знал какой драйвер искать */  
REPORTDATABASENAME=ms-r8  
/* имя базы */  
REPORTDATABASESERVER=8.8.8.8  
REPORTDATABASEPORT=1433  
/* ip и порт по которому mssql-server ждёт подключения */  
/* если менялся при настройке mssql-server указать соответствующий */  
REPORTDATABASELOCALE=RU_RU.CP1251  
REPORTWEBSERVERPATH = /usr/local/totvswebserver/server/webapps/webviewer/  
/* путь до каталога webviewer */
```

8. Перед тестовым подключением проверить состояние всех служб командой **systemctl**.

Smartclient

Для настройки Smartclient следует выполнить следующие действия:

1. Выберите в каталоге Smartclient клиент для вашей операционной системы.
2. Распакуйте архив с клиентом в произвольное место на жестком диске.
3. В каталоге **smartclient** для запуска с русской локализацией необходимо переименовать **smartclient.exe** в **npsmartclient.exe**.
4. Создать файл **npsmartclient.ini** и заполнить его следующим образом:

```
[config]
lastmainprog=sigaadv
envserver=
/* написать название блока environmet из appserver.ini */

[drivers]
active=ma3
/* указать название блока с server, port, Environment */
[ma3]
server=8.8.8.8
/* ip, указанный в appserver.ini
“[LockServer]
Enable=1
Server=8.8.8.8
Port=12100” */
port=12100
Environment=ma3-appliance
/* название блока среды, указанное в appserver.ini */
```