



// Руководство по установке

БАЗИС 12.10



Содержание

1	Установка программного обеспечения	4
1.1	Установка сервера приложений (на примере Tomcat 8.5)	4
1.1.1	В ОС Windows.....	4
1.1.2	В ОС Unix / Linux	5
1.2	Настройка сервера приложений (на примере Tomcat 8.5).....	7
1.2.1	Дополнительно: включение сжатия	7
1.2.2	Дополнительно: настройка HTTPS	8
1.2.3	Дополнительно: Настройка другого контекста.....	9
1.2.4	Дополнительно: настройка Apache для статического содержимого	10
1.3	Установка сервера лицензий	10
1.3.1	В ОС Windows.....	10
1.3.2	В ОС Unix / Linux	11
1.4	Настройка сервера лицензий	12
1.4.1	Открытие веб-интерфейса сервера лицензий	12
1.4.2	Проверка статуса сервера лицензий	13
1.4.3	Настройка \ удаление службы Windows для сервера лицензий	14
1.4.4	Перезапуск сервера лицензий	15
1.4.5	Загрузка лицензии.....	16
1.4.6	Дополнительно: администрирование управления доступом к веб-интерфейсу	16
1.5	Установка приложения БАЗИС	18
1.5.1	Настройки для функции поиска в системе БАЗИС	21
1.5.2	Дополнительно: интеграция LDAP	22
1.5.3	Дополнительно: активирование API	24
1.5.4	Дополнительно: шифрование файла CommandConfig.xml	24
1.5.5	Дополнительно: шифрование паролей в базе данных	24
1.5.6	Дополнительно: каталоги сообщений для других языков	26
1.5.7	Дополнительно: вызов локального клиента Putty для Telnet.....	26
1.5.8	Дополнительно: изменение языка для страницы входа в систему и языка пользователя	27
1.5.9	Дополнительно: настройка другого контекста	28
1.5.10	Дополнительно: функционирование приложения БАЗИС со шлюзом SSO (системой единого входа)	28
2	Дополнительные варианты установки	29
2.1	Вариант 2.....	29
2.1.1	Настройки для мультисерверной архитектуры (кластера).....	29
2.2	Вариант 3.....	29
2.2.1	Настройка сервера Apache.....	30
2.2.2	Дополнительно: распределение приложения БАЗИС между Tomcat и Apache	32
2.3	Вариант 4.....	33
2.3.1	Настройка распределителя нагрузки (Load Balancer)	33
2.3.2	Настройка функции BALANCER_SESSION_STICKY	34
3	Настройка дополнительных модулей	35
3.1	Инициализация метасхемы БАЗИС	35
3.2	Почтовый сервер по умолчанию	36
3.3	Редакторы запросов	37
3.4	Радиосети.....	38
3.5	Управление ЦОД.....	39
3.5.1	Активирование модулей для управления ЦОД.....	39
3.5.2	Установка.....	39
3.6	Телеком.....	41
3.7	Наряды.....	41



3.8	3D план помещения	42
3.8.1	Активирование модуля «3D план помещения»	42
3.9	SOAP API	42
3.10	Запросы на прокрутку (для интеграционного слоя)	43
3.11	Клиентское расширение БАЗИС	44
4	Обновление существующих систем	45
4.1	Обновление программы - директория приложения БАЗИС	45
4.2	Обновление программы - пакеты исправлений и заплат	47
5	Список изображений	49
6	Выходные данные	50

1 Установка программного обеспечения

Стандартная установка

В стандартной установке Apache Tomcat выступает в качестве веб-сервера и соединяется с сервером баз данных, используемым для хранения данных системы БАЗИС. При этом оба сервера могут находиться как на одном общем компьютере, так и на двух отдельных.

В дальнейшем описывается процесс установки приложения БАЗИС с использованием стандартного варианта.

1.1 Установка сервера приложений (на примере Tomcat 8.5)

1.1.1 В ОС Windows

Порядок выполнения операций

Установка выполняется при помощи специальной программы (мастера установки):

- Выполните файл **apache-tomcat-x.y.z.exe** для запуска мастера установки.
- Следуйте указаниям на экране.
- Активируйте на шаге установки **Choose Components** следующую опцию: **Tomcat** → **Service Startup**.
- Измените путь в соответствии с Вашим системным окружением. Для директории установки в дальнейшем используется обозначение **<\$TOMCAT_HOME>**.



Внимание

В названии пути не должно быть пробелов.

- В ходе установки на шаге **Configuration Options** Вы можете изменить порт, который слушает Tomcat.



Внимание

Если на шаге **Configuration Options** не указать пароль, то вход в **Tomcat Web Application Manager** будет возможен без пароля под пользователем **admin**.

- В параметре Java Virtual Machine выберите путь к установленной среде JRE (jre\bin\server\jvm.dll).
- Процесс настройки можно запустить из главного меню программы **Configure Tomcat** (на вкладке Java).



Примечание / рекомендация

После установки измените конфигурацию Tomcat, установив для параметра Initial / Maximal Memory Pool (исходный / максимальный пул памяти) значение 512 МБ / 1024 МБ. В зависимости от доступного объема оперативной памяти можно выставить более высокие или низкие значения (но не менее 256 МБ). Если используется Web Service API (интерфейс прикладных программ для веб-служб), значение максимального пула памяти должно быть не менее 2048 МБ. Для 32-разрядной операционной системы в качестве максимального значения можно указать 1536 МБ.

- Добавьте в текстовое поле **Java Options** следующую запись:

-XX:MaxMetaspaceSize=512m

В версии Java 1.8 параметр MaxPermSize был переименован в MaxMetaspaceSize.

- Добавьте в поле **Java Classpath** запись **<\$TOMCAT_HOME>\sharedVib**.

- Конфигурирование значений памяти возможно также через реестр. Для этого необходимо изменить значения параметров REG_DWORDS JvmMs и JvmMx в HKEY_LOCAL_MACHINE\SOFTWARE\Apache Software Foundation\Procrun 2.0\Tomcat5\Parameters\Java на желаемые значения.
- Откройте файл $\\$TOMCAT_HOME\conf\catalina.properties$ для редактирования. Добавьте в строку «**shared.loader**» значение **$\${catalina.base}/shared/lib/*.jar$** .

```
shared.loader=${catalina.base}/shared/lib/*.jar
```

- Скопируйте директорию **shared/lib**, содержащуюся в пакете сервера лицензий, в **$\$TOMCAT_HOME$** .
- Добавьте путь до директории $\\$TOMCAT_HOME\shared\lib$ к переменной среды **Path**:
- Откройте окно **Свойства системы** Вашей операционной системы:
- Откройте режим редактирования переменных среды на вкладке **Дополнительно**, щелкнув по кнопке **Переменные среды**.
- Открывается окно **Переменные среды**:
- Выберите переменную **Path** в области **Системные переменные**, щелкнув по соответствующей строке.
- Щелкните по кнопке **Редактировать**.
- Открывается окно **Изменение системной переменной**.
- Добавьте путь к папке «;$\\$TOMCAT_HOME\shared\lib$» (например, **C:\Command\Tomcat\Shared\Lib**) в поле **Значение переменных**.



Внимание

Помните, что пользовательские переменные отличаются от системных переменных.

- Щелкните по кнопке **ОК** для подтверждения выполненных изменений.

1.1.2 В ОС Unix / Linux

Порядок выполнения операций

- Разархивируйте файл **apache-tomcat-x.y.z.tar.gz** в желаемую директорию (называемую в дальнейшем **$\$TOMCAT_HOME$**).
- Скопируйте директорию **shared/lib**, входящую в пакет сервера лицензий, в **$\$TOMCAT_HOME$** .
- Откройте файл **$\$TOMCAT_HOME/conf/catalina.properties$** для редактирования. Добавьте в строку **shared.loader** значение **$\${catalina.base}/shared/lib/*.jar$** .

```
shared.loader=${catalina.base}/shared/lib/*.jar
```

- Сделайте файл **$\$TOMCAT_HOME/shared/lib/librlm1002.so$** исполняемым.

```
chmod +x librlm1002.so
```

- Скопируйте файл **setenv.sh**, входящий в пакет сервера лицензий, в директорию **$\$TOMCAT_HOME/bin$** .



- Преобразуйте файлы **\$TOMCAT_HOME/bin/shutdown.sh**, **startup.sh** и **setenv.sh** из формата DOS в UNIX.

В ОС Unix: dos2unix <имя_файла.sh> <имя_файла.sh>

В ОС Linux: dos2unix <имя_файла.sh>

- Измените в директории **\$TOMCAT_HOME/bin** статус файлов **shutdown.sh**, **startup.sh**, **catalina.sh**, **setenv.sh** и **setclasspath.sh** на исполняемый.

```
chmod +x shutdown.sh startup.sh catalina.sh setclasspath.sh setenv.sh
```

- Добавьте в начало файла **catalina.sh** следующие записи. Измените пути в соответствии с Вашим системным окружением:

```
TOMCAT_HOME=/pfad/zur/tomcat-installation
CATALINA_HOME=/pfad/zur/tomcat-installation
JRE_HOME=/pfad/zur/sdk-installation
JAVA_HOME=/pfad/zur/sdk-installation
LANG=de_DE
JAVA_OPTS="-Xms512M -Xmx2048M -XX:MaxMetaspaceSize=512m"
export CATALINA_HOME TOMCAT_HOME JAVA_HOME JRE_HOME LANG
JAVA_OPTS
```



Примечание / рекомендация

В системах без сконфигурированного графического вывода (X-сервер) переменная **JAVA_OPTS** должна быть задана следующим образом:

```
JAVA_OPTS="-Xms512M -Xmx2048M -XX:MaxMetaspaceSize=512m -
Djava.awt.headless=true"
```

В версии Java 1.8 параметр **MaxPermSize** был переименован в **MaxMetaspaceSize**.

В зависимости от имеющейся в распоряжении оперативной памяти, Вы можете установить другие значения для параметров **Xms** (исходное) и **Xmx** (максимальное значение памяти для сервера приложений), но не менее 256 МБ. Если используется Web Service API (интерфейс прикладных программ для веб-служб), то значение максимального пула памяти должно быть не менее 1024 МБ.

- Если необходимо, в файле **\$TOMCAT_HOME/conf/server.xml** можно изменить порт HTTP.

Пример:

```
<!-- Define a non-SSL HTTP/1.1 Connector on port 8080 -->
<Connector port="8080" maxHttpHeaderSize="8192"
```



Примечание / рекомендация

Для того чтобы Tomcat работал на порте 80, служба должна быть запущена с правами **root**.

1.2 Настройка сервера приложений (на примере Tomcat 8.5)

Порядок выполнения операций

- Добавьте в конец файла `$TOMCAT_HOME/conf/server.xml` перед записью `</Host>` следующие строки.

```
<Context path="" docBase="command" debug="0" />
<Context path="/axis" docBase="command/axis" debug="0" reloadable="false"
crossContext="true" swallowOutput="true"/>
<Context path="/wmicons" docBase="C:\command_app\wmicons" debug="0"/>
<Context path="/doc" docBase="C:\command_app\doc" debug="0" />
```

1.2.1 Дополнительно: включение сжатия

Порядок выполнения операций

Tomcat может сжимать передаваемые клиенту данные, в результате чего уменьшаются объем передаваемых данных и требуемая ширина полосы пропускания. Сжатие рекомендуется включать при малой скорости передачи данных или небольшой ширине полосы пропускания между клиентом и сервером, как это, например, бывает при передаче данных через сеть WAN.

Для включения сжатия необходимо установить в файле `$TOMCAT_HOME/conf/server.xml` для элемента `<Connector>` следующие дополнительные атрибуты для порта HTTP.

```
compression="on"
compressableMimeType="text/html,text/xml,text/plain"
compressionMinSize="2048"
noCompressionUserAgents="gozilla, traviata"
```

Пример полного элемента `<Connector>`:

```
<Connector port="8080" maxThreads="150" minSpareThreads="25"
maxSpareThreads="75" enableLookups="false" redirectPort="8443" acceptCount="100"
debug="0" connectionTimeout="20000" disableUploadTimeout="true" compression="on"
compressableMimeType="text/html,text/xml,text/plain" compressionMinSize="2048"
noCompressionUserAgents="gozilla, traviata"/>
```

1.2.2 Дополнительно: настройка HTTPS

Протокол HTTPS используется для шифрования потока данных между клиентом (браузером) и веб-сервером (Tomcat). HTTPS позволяет шифровать поток данных, предоставляя тем самым надежную защиту от прослушивания информации. Если у клиента и веб-сервера отсутствует сертификат для посещаемого сайта, браузеры классифицируют такой сайт как ненадежный и предупреждают пользователя о возможной опасности, исходящей от такого сайта. Для предотвращения подобной опасности используется сертификат. Сертификат предоставляется органом сертификации и подтверждает подлинность веб-сайта или его владельца. Это означает, что пользователь может быть уверен, что лицо или фирма действительно существует и что в соединении между клиентом и сервером отсутствуют промежуточные узлы. В Tomcat такой сертификат запакван в файле Keystore и может быть настроен соответствующим образом.



Настройка файла Keystore описывается в руководстве по установке Tomcat. Файл Keystore - это файл, создаваемый Java и содержащий сертификат для сертификации SSL. За создание и сертификацию сертификата SSL, а также за создание файла Keystore отвечает сам заказчик.

По умолчанию при установке Tomcat для порта HTTPS Connector параметр Keystore не устанавливается. В зависимости от используемого источника установки Connector можно закомментировать или раскомментировать. В нашем примере мы исходим из того, что он закомментирован.

Порядок выполнения операций

Все действия по настройке HTTPS должны выполняться на Tomcat в файле server.xml. Как найти путь к файлу, описано в разделе *НАСТРОЙКА СЕРВЕРА ПРИЛОЖЕНИЙ*.

Порт HTTPS Connector из файла server.xml перед настройкой:

```
<!-- Define a SSL HTTP/1.1 Connector on port 8443 -->
<!--
<Connector port="443" protocol="HTTP/1.1" SSLEnabled="true"
maxHttpHeaderSize="8192" maxThreads="150" minSpareThreads="25"
maxSpareThreads="75" enableLookups="false" disableUploadTimeout="true"
acceptCount="100" scheme="https" secure="true" clientAuth="false" sslProtocol="TLS"/>
-->
```

Для настройки необходимо удалить знаки комментариев и добавить запись **keystoreFile**. Для защиты Keystore паролем необходимо добавить запись **keystorePass**.



Примечание / рекомендация

Запись **keystoreType** является дополнительной функцией, которая требуется, только если KeyStore не является файлом KeyStore (например, pkcs12).

Порт HTTPS Connector из файла server.xml после настройки:

```
<!-- Define a SSL HTTP/1.1 Connector on port 8443 -->
<Connector port="443" protocol="HTTP/1.1" SSLEnabled="true"
maxHttpHeaderSize="8192" maxThreads="150" minSpareThreads="25"
maxSpareThreads="75" enableLookups="false" disableUploadTimeout="true"
acceptCount="100" scheme="https" secure="true" clientAuth="false" sslProtocol="TLS"
keystoreType="PKCS12"
keystoreFile="/путь/до/файла/.keystore" keystorePass="xxxxxxx" />
```

Настройка переадресации HTTPS

В базе данных можно управлять реакцией системы БАЗИС на запрос HTTP / HTTPS. При этом можно задать, будут ли допускаться оба типа запросов (HTTP и HTTPS) или, из соображений безопасности, только запросы HTTPS.

Для этого в настройках базы данных доступны следующие опции:

Значение	Описание
request_dependent	(По умолчанию) При этой настройке решение об использовании SSL принимается на основе используемого вызова URL (HTTP или HTTPS).

SSL_only

При этой настройке для коммуникации используется только HTTPS, в результате чего все данные передаются в зашифрованном виде. Если пользователь вызывает приложение БАЗИС через HTTP, то выполняется автоматическая переадресация на HTTPS.

Следующая команда SQL позволяет задать / изменить в базе данных настройку в схеме **command** посредством SQL*Plus:

```
update STFCFG_SETTING
  set VALUE = '<value>'
where NAME = 'SYS_HTTP_SECURITY' and SW_UNIT='Login'
```

Пример:

```
update STFCFG_SETTING
  set VALUE = 'SSL_only'
where NAME = 'SYS_HTTP_SECURITY' and SW_UNIT='Login'
```

1.2.3 Дополнительно: Настройка другого контекста

Если приложение БАЗИС используется в другом контексте, необходимо дополнить атрибут **path** элемента **<Context>** в файле **\$TOMCAT_HOME\conf\server.xml** собственным контекстом. Например, если приложение БАЗИС используется в контексте **/app/command**, то запись будет выглядеть следующим образом:

```
<Context path="/app/command" docBase="command" debug="0"/>
<Context path="/app/command/axis" docBase="command/axis" debug="0"
  reloadable="false" crossContext="true" swallowOutput="true"/>
```

Кроме того, для приложения БАЗИС необходимо выполнить шаги, описанные в разделе 1.5.9.



Примечание / рекомендация

Контекст является составной частью лицензии URL экземпляра системы БАЗИС. При изменении контекста необходимо установить новую лицензию URL.

1.2.4 Дополнительно: настройка Apache для статического содержимого

Если для работы со статическими файлами используется сервер Apache, то необходимо соответствующим образом изменить конфигурацию, чтобы, например, активировать глубинные ссылки. Данную операцию можно пропустить, если сервер Apache будет использоваться только для перенаправления запросов к серверу приложений (например, в качестве прокси-сервера или распределителя нагрузки).

Как правило, все запросы к **/html/**** на сервере приложений принимает сервлет «HtmlFrontendServlet» и перенаправляет их к файлу **index.html**, если требуемый файл не удается найти. Маршрутизатор фронтенда автоматически распознает указанный путь и применяет внутреннюю маршрутизацию.

В этом случае сервер Apache должен перенаправлять также все **/html-**запросы к файлу **index.html**, если требуемый файл не удается найти. Для этого необходимо установить и активировать модуль **mod_rewrite**.



Модуль `mod_rewrite` сервера Apache:

https://httpd.apache.org/docs/current/mod/mod_rewrite.html

Создайте файл `.htaccess` в директории `html` и добавьте к нему следующие записи:

```
RewriteEngine On
RewriteCond %{DOCUMENT_ROOT}%{REQUEST_URI} -f [OR]
RewriteCond %{DOCUMENT_ROOT}%{REQUEST_URI} -d
RewriteRule ^ - [L]
RewriteRule ^ index.html
```

Данный код можно также добавить в конфигурацию Apache.

1.3 Установка сервера лицензий

1.3.1 В ОС Windows

Установка

- Разархивируйте входящий в пакет поставки файл **`licensesrv_1002.zip`** в желаемую директорию, называемую в дальнейшем **`$LICENSE_SRV_HOME`**.
- Выберите директорию по умолчанию для хранения файлов лицензий (например, **`$LICENSE_SRV_HOME/licenses`**). Позднее при настройке посредством веб-интерфейса Вам будет необходимо прописать на сервере лицензий информацию об этой директории.

Порядок выполнения операций

- Исполните файл **`rlm.exe`**.
- Выполняется запуск сервера лицензий:

```

11/26 13:20 <rlm> RLM License Server Version 9.4BL2
      Copyright (C) 2006-2012, Reprise Software, Inc. All rights reserved.
11/26 13:20 <rlm> License server started on PC-FKN-2011
11/26 13:20 <rlm> Server architecture: x86_w3
11/26 13:20 <rlm> License files:
11/26 13:20 <rlm>     fnt_test_customer.lic
11/26 13:20 <rlm>
11/26 13:20 <rlm> Web server starting on port 5054
11/26 13:20 <rlm> Using TCP/IP port 2375
11/26 13:20 <rlm> Starting ISU servers:
11/26 13:20 <rlm>     ... fnt on port 50618
11/26 13:20 <rlm> New thread created to watch ISU fnt
11/26 13:20 <fnt> RLM License Server Version 9.4BL2 for ISU "fnt"
11/26 13:20 <fnt> Settings from RLM Version 9.4BL2 for ISU "fnt"
11/26 13:20 <fnt> Server architecture: x86_w3
      Copyright (C) 2006-2012, Reprise Software, Inc. All rights reserved.
      RLM contains software developed by the OpenSSL Project
      for use in the OpenSSL Toolkit (http://www.openssl.org)
      Copyright (c) 1998-2008 The OpenSSL Project. All rights reserved.
      Copyright (c) 1995-1998 Eric Young (eay@cryptsoft.com) All rights reserved.
11/26 13:20 <fnt> WARNING: cline_d01 v9.03 license expires within 14 days
11/26 13:20 <fnt> WARNING: cbase_d01 v9.03 license expires within 14 days
11/26 13:20 <fnt> WARNING: telco_d01 v4.5 license expires within 14 days
11/26 13:20 <fnt> WARNING: cline_d02 v9.03 license expires within 14 days
11/26 13:20 <fnt> WARNING: cbase_d02 v9.03 license expires within 14 days
11/26 13:20 <fnt> WARNING: telco_d02 v4.5 license expires within 14 days
11/26 13:20 <fnt>
11/26 13:20 <fnt> Server started on PC-FKN-2011 (hostid: ANY) for:
11/26 13:20 <fnt>     cline_d01 cbase_d01 telco_d01 cline_d02 cbase_d02
11/26 13:20 <fnt>     telco_d02 url_900bfac515c720bf747a0ec1c76c901d url_531de
4f8287092202660a821807a2cb4e
11/26 13:20 <fnt>     url_8c382082503da367321b1934d6f44f7e url_35aa81a97297d12
6ce0d289eb93622a8
11/26 13:20 <fnt>     url_9825c2a542dd888e55b9b0e06b04f672 language_de_de
11/26 13:20 <fnt>     language_en_us
11/26 13:20 <fnt>
11/26 13:20 <fnt> License files:
11/26 13:20 <fnt>     fnt_test_customer.lic
11/26 13:20 <fnt>

```

Рисунок 1-1 rlm.exe

На одном из серверов сервер лицензий должен быть настроен как служба. Данную службу можно создать в веб-интерфейсе сервера лицензий (смотрите раздел *НАСТРОЙКА И УДАЛЕНИЕ СЛУЖБЫ WINDOWS ДЛЯ СЕРВЕРА ЛИЦЕНЗИЙ*). Для этого необходимо исполнить файл rlm.exe с правами администратора.

1.3.2 В ОС Unix / Linux

Установка

- Разархивируйте входящий в пакет поставки файл **licensesrv_1002.zip** в желаемую директорию, называемую в дальнейшем \$LICENSE_SRV_HOME.
- Преобразуйте файл **rlm.sh** из формата DOS в UNIX.
В ОС Unix: dos2unix <имя_файла.sh> <имя_файла.sh>
В ОС Linux: dos2unix <имя_файла.sh>
- Сделайте файлы rlm, rlm-down, rlm-util и rlm.sh исполняемыми.

```
chmod +x rlm rlm-down rlm-util rlm.sh
```

- Выберите директорию по умолчанию для хранения файлов лицензий (например, **\$LICENSE_SRV_HOME/licenses**).

Перед запуском сервера лицензий необходимо изменить файл **\$LICENSE_SRV_HOME/rlm.sh**.



- В параметре ***rlmdir*** необходимо задать директорию для сервера лицензий:

```
rlmdir=$LICENSE_SRV_HOME
```

- Если она уже была настроена на предыдущем шаге, место хранения можно изменить при помощи параметра ***licpath***.

```
licpath=$rlmdir/licenses
```

Порядок выполнения операций

- Исполните файл ***rlm.sh*** с параметром ***start***.

```
./rlm.sh start
```

- ✓ Выполняется запуск сервера лицензий.

- Исполните файл ***rlm.sh*** с параметром ***stop***.

```
./rlm.sh stop
```

- ✓ Сервер лицензий останавливается.

- Исполните файл ***rlm.sh*** с параметром ***restart***.

```
./rlm.sh restart
```

- ✓ Сервер лицензий сначала останавливается, а затем снова запускается.

1.4 Настройка сервера лицензий

1.4.1 Открытие веб-интерфейса сервера лицензий

Сервер лицензий обладает веб-интерфейсом, с помощью которого можно осуществлять контроль за сервером.

Порядок выполнения операций

- Откройте в браузере следующий адрес (порт 5054), чтобы зайти в веб-интерфейс сервера лицензий:

```
http://<LICENSE_SERVER_HOSTNAME>:5054/home.asp
```

Пример адреса:

```
http://localhost:5054/home.asp
```

Пользователь по умолчанию и управление доступом описываются в разделе **ДОПОЛНИТЕЛЬНО: АДМИНИСТРИРОВАНИЕ УПРАВЛЕНИЯ ДОСТУПОМ К ВЕБ-ИНТЕРФЕЙСУ**.

- ✓ Открывается веб-интерфейс сервера лицензий:

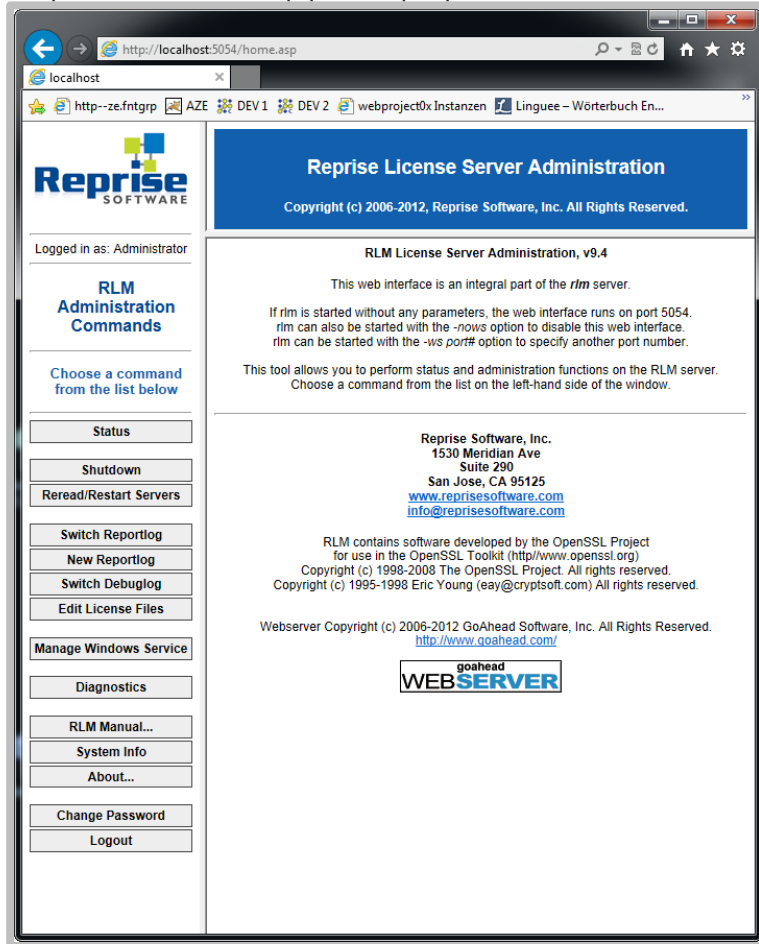


Рисунок 1-2 Веб-интерфейс сервера лицензий

1.4.2 Проверка статуса сервера лицензий

При помощи функции **Status** («Статус») в левой части панели инструментов можно отобразить статус сервера лицензий. В открывшейся области данных показываются запущенные в текущий момент экземпляры серверов с установленными на них лицензиями:

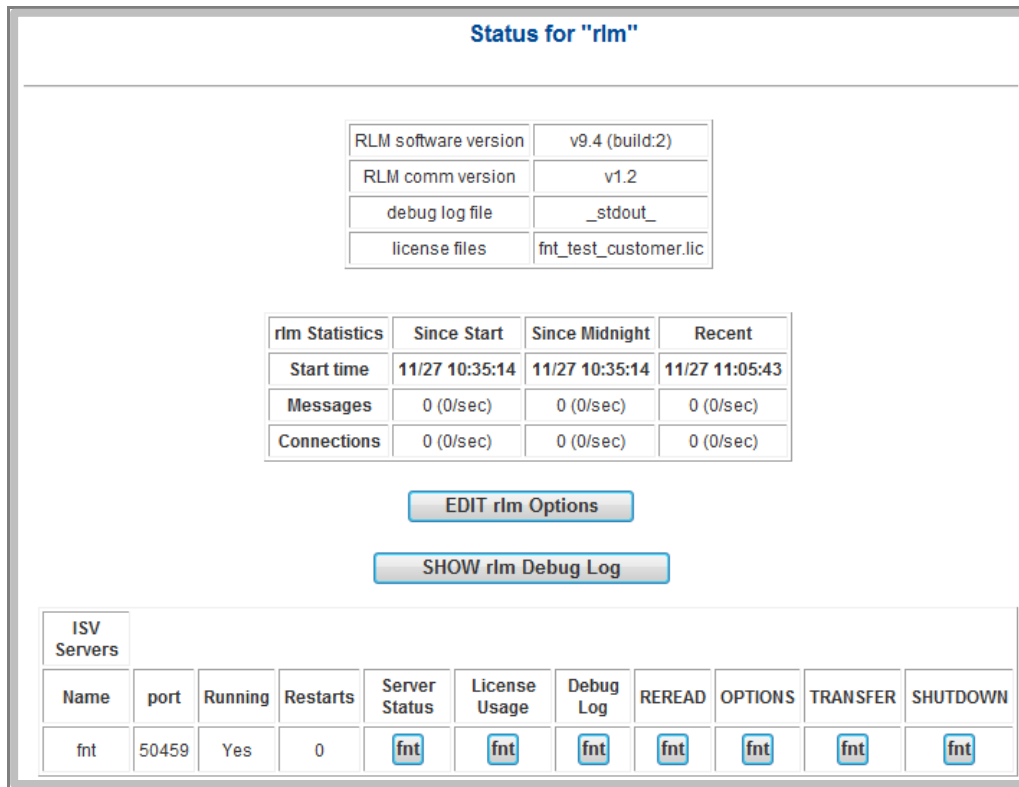


Рисунок 1-3 Статус rlm

Порядок выполнения операций

- Щелкните в веб-интерфейсе на панели инструментов слева по функции **Status** («Статус»).
- ✓ Открывается область данных, отображающая статус сервера лицензий.

1.4.3 Настройка \ удаление службы Windows для сервера лицензий

Для сервера лицензий службу Windows можно настроить в веб-интерфейсе. Для этого необходимо исполнить файл **rlm.exe** с правами администратора.

Порядок выполнения операций

- Щелкните в веб-интерфейсе на панели инструментов слева по функции **Manage Windows Service** («Управление службами Windows»).

- ✓ Открывается область данных для создания службы Windows для сервера лицензий:

Рисунок 1-4 Управление службами Windows для сервера лицензий RLM

- Выберите в поле **Service Action** («Действия с сервером»), что необходимо сделать со службой: создать службу Windows (**INSTALL service**, «Установить службу») или удалить (**REMOVE service**, «Удалить службу»).
- Щелкните по кнопке **Apply** («Применить»), чтобы выполнить соответствующую операцию.
- ✓ Если в поле **Webserver port#** («Порт веб-сервера») не задается новый порт, то при настройке службы для вызова веб-интерфейса сохраняется предыдущий порт.
- *Дополнительно:* Измените в поле **Server Licence File Directory** («Директория для файла сервера лицензий») место для хранения файлов лицензий.

1.4.4 Перезапуск сервера лицензий

Сервер лицензий можно перезапустить посредством веб-интерфейса. Перезапуск может, например, потребоваться после сохранения нового файла лицензии.

Порядок выполнения операций

- Щелкните в веб-интерфейсе в левой части панели инструментов по функции **Reread/Restart Servers** («Считать заново / перезапустить серверы»).
- ✓ Открывается область данных с параметрами для перезапуска:

Рисунок 1-5 Опции для перезапуска

- Выберите в поле **ISV** опцию **СДИ**.



- Щелкните по кнопке **REREAD/RESTART** («Считать заново / перезапустить»), чтобы выполнить перезапуск.
- ✓ На экране отображается подтверждение перезапуска. **Reread command sent to fnt** («Команда на повторное считывание отправлена в СДИ»).

1.4.5 Загрузка лицензии

Порядок выполнения операций

- Сохраните файл лицензии на сервере лицензий в директории, указанной во время установки.
- Перезапустите сервер лицензий (смотрите раздел *ПЕРЕЗАПУСК СЕРВЕРА ЛИЦЕНЗИЙ*), чтобы сохранить изменения.

1.4.6 Дополнительно: администрирование управления доступом к веб-интерфейсу

Доступом к веб-интерфейсу сервера лицензий можно управлять через файл **rlm.pw**. Данный файл находится в установочной директории сервера лицензий и должен быть защищен от неавторизованного доступа. В данном файле для каждого пользователя предусмотрена строка, в которой зафиксирована вся информация о доступе в следующем формате

```
//имя пользователя:пароль:список прав//
```

Данные в профиле доступа разделены двоеточием.

Область **//имя пользователя//** содержит регистрационное имя пользователя для входа в систему. Данное имя не должно содержать разделительных знаков (двоеточия).

Если область **//пароль//** не содержит никаких данных (**//имя пользователя::список прав//**), вход пользователя в систему возможен без указания пароля. Если у пользователя есть пароль, заданный посредством веб-интерфейса, он отображается в зашифрованном виде в области **//пароль//**.

В состоянии при поставке пользователю по умолчанию **Administrator** присвоен пароль **12345**. Данный пароль можно изменить в веб-интерфейсе.

В области **//список прав//** задаются права доступа для пользователя. Доступны следующие полномочия:



Полномочие	Значение	Дополнительная информация
all	Собирательное полномочие - содержит все другие полномочия.	
edit_meter	Позволяет редактировать порядок нумерации для функций измерений.	Автоматически активирует полномочие «status».
edit_options	Позволяет редактировать файлы параметров ISV.	Автоматически активирует полномочие «status».
logfiles	Позволяет администрировать лог-файлы в веб-интерфейсе.	
remove	Позволяет пользователю удалять лицензию из текущего процесса.	Автоматически активирует полномочие «status».
reread	Позволяет пользователю заново загружать лицензии.	
shutdown	Позволяет пользователю выключать сервер лицензий посредством веб-интерфейса.	Автоматически активирует полномочие «status».
status	Позволяет пользователю отображать информацию о статусе сервера лицензий посредством веб-интерфейса.	

Пример:

```
Administrator:<зашифрованный пароль>:all  
Guest::status
```

Администрирование управления доступом через файл *rlm.opt*

Управлять доступом к веб-интерфейсу сервера лицензий также можно через файл *rlm.opt*. Данный файл должен находиться в установочной директории сервера лицензий. Управление доступом выполняется посредством выдачи полномочий одному или нескольким IP-адресам или IP-сетям.

Пример:

ACTIVATE off

→ Данная строка деактивирует кнопку **Activate License** («Активировать лицензию»).

INCLUDE status internet 192.168.1.* 192.168.2.* 192.168.3.*

→ Данная строка позволяет посредством веб-интерфейса выполнять запросы о статусе для IP-сетей 192.168.1.*, 192.168.2.* и 192.168.3.*.

INCLUDE reread internet 192.168.2.155

→ Данная строка позволяет заново загружать лицензии для IP-адреса 192.168.2.155.



INCLUDE shutdown internet 192.168.2.155

→ Данная строка позволяет выключать сервер лицензий для IP-адреса 192.168.2.155.

INCLUDE edit_options internet 192.168.2.155

→ Данная строка позволяет управлять сервером лицензий для IP-адреса 192.168.2.155.

INCLUDE edit_rlm_options internet 192.168.2.155

→ Данная строка позволяет редактировать файл rlm.opt для IP-адреса 192.168.2.155.

INCLUDE remove internet 192.168.2.155

→ Данная строка позволяет удалять лицензии из текущего процесса для IP-адреса 192.168.2.155.

INCLUDE manage_service internet 192.168.2.155

→ Данная строка позволяет управлять службой лицензий для IP-адреса 192.168.2.155.

1.5 Установка приложения БАЗИС

Порядок выполнения операций

Установка

- Распакуйте архив `command_xx.yy.zz_mod_x.zip` в директорию **`$TOMCAT_HOME\webapps\`**
- Система **БАЗИС** не должна использоваться совместно с другими веб-приложениями (например, **FNT StagingArea**) внутри одного и того же экземпляра сервера приложений. В зависимости от окружения для этих целей можно использовать различные серверы, хотя это и не является обязательным условием.

Настройка



Примечание / рекомендация

Начиная с версии 12.6.4, все значения в файле CommandConfig.xml можно заменить на подстановочные знаки системных свойств Java. Системные свойства Java можно задать в файле $\\$TOMCAT_HOME\conf\catalina.properties$

```
...
myProperty=8080
commandFiles=/var/opt/command
...
```

или в переменной JAVA_OPTS.

```
JAVA_OPTS="-DmyProperty=8080 -DcommandFiles=/var/opt/command ..."
```

В файле CommandConfig.xml свойства можно задать следующим образом. Также можно использовать несколько свойств в одном значении, а подстановочные знаки вместе с обычными значениями.

```
<property name="apachePort" value="\${myProperty}"/>
<property name="wmlconsHome" value="\${commandFiles}/wmlcons"/>
<property name="example"
value="\${example.property.1}/static/\${example.property.2}"/>
```

Замена подстановочных знаков возможна во всех атрибутах «value» файла CommandConfig.xml, а также для значений узлов XML «dbserver», «ldapserver», «searchindex», «messagequeue» и «language».

Внесите в файл $\$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes\CommandConfig.xml$ следующие изменения:

- Измените запись для **apachePort** в соответствии со значением для порта Tomcat. Пример:

```
<property name="apachePort" value="8080"/>
```

Если система БАЗИС работает через HTTPS, то необходимо изменить запись **apachePortSSL** соответствующим образом.

Пример HTTPS:

```
<property name="apachePortSSL" value="443"/>
```

- Задайте при необходимости постоянный протокол передачи данных. Например, это может понадобиться, если прокси изменяет протокол HTTP на HTTPS. Пример:



```
<property name="constantProtocol" value="https" />
```

С данной настройкой приложение БАЗИС получает указание выполнять запросы через HTTPS, даже если прокси передает их дальше серверу как HTTP.

- Измените запись **wmlconsHome** в соответствии с Вашими требованиями.
Пример:

```
<property name="wmlconsHome" value="/pfad/zur/tomcat-  
installation/webapps/command/wmicons"/>
```

- Присвойте экземпляру уникальный идентификатор (**APP_SERVER_ID**). Данный идентификатор не должен использоваться ни для одного из других экземпляров. Длина ограничена 14-ю знаками.
Пример:

```
<property name="applicationServerId" value="APP_SERVER_ID"/>
```

- Для того чтобы приложение БАЗИС могло обмениваться данными с сервером лицензий, необходимо настроить его адрес.
В односерверном режиме работы достаточно указать запись **localhost**, если сервер лицензий работает на сервере приложений. В мультисерверном режиме работы это невозможно.
Пример:

```
<property name="licServerHost" value="LICENSE_SERVER_HOSTNAME"/>
```

- Указание идентификатора системы из файла лицензии соответствует суффиксу имени лицензии, например, для cbase_fkx идентификатором системы будет fkx. Идентификатор системы идентичен для всех лицензий.

```
<property name="licSystemId" value="xxx"/>
```

- Можно также указать дополнительные серверы лицензий в качестве резервных серверов. Они будут использоваться системой БАЗИС, если сервер лицензий с идентификатором **default** (по умолчанию) будет недоступен. Для этого необходимо раскомментировать и соответствующим образом изменить шаблон с идентификатором **backup_01**. Для создания дополнительных резервных серверов лицензий данный шаблон можно скопировать нужное количество раз и присвоить серверам уникальные идентификаторы.

```
<server id="backup_01">
    <property name="licSystemId" value="xx-0"/>
    <property name="licServer" value="FNT_RLM"/>
    <property name="licServerHost" value="backup_licence_server_host"/>
    <property name="licServerPort" value="2375"/>
    <property name="licRlmLib" value="rlm1002"/>
</server>
```

1.5.1 Настройки для функции поиска в системе БАЗИС

1.5.1.1 Активирование серверных задач

Управление индексированием осуществляется посредством специальной серверной задачи. Поэтому серверные задачи должны быть активированы в файле **CommandConfig.xml**. Для этого необходимо установить для параметра **serverJobs** значение **enabled**.

Затем необходимо изменить путь до индекса:

```
<searchindex maxParallelThreads="4" commitThreshold="250">
  <index id="default" path="[wmlconsHome]\searchindex" />
</searchindex>
```

1.5.1.2 Дополнительно: настройка индексируемых языков

При помощи параметра **SYS_EASY_SEARCH_LANGUAGES** можно задать, какие из лицензированных языков должны включаться в индекс. Для указания нескольких языков используется символ «|». В настоящее время можно установить следующие значения:

- de_DE для немецкого языка
- en_US для английского языка
- ru_RU для русского языка
- fr_FR для французского языка

1.5.1.3 Дополнительно: настройка директории для хранения индекса

Посредством узла **searchindex** в файле **CommandConfig.xml** можно изменить текущий путь до директории, в которой хранится индекс.

Для создания нескольких синхронизируемых индексов внутри узла **searchindex** можно создать дополнительный подузел. Данный способ используется для распределения резервных индексов на различные системы, чтобы повысить отказоустойчивость индекса; в этом случае отпадает необходимость создания индекса заново при появлении проблем с доступом.

1.5.1.4 Дополнительно: настройка количества параллельных процессов для индексирования

В узле **searchindex** в файле **CommandConfig.xml** при помощи параметра **maxParallelThreads** можно задать максимальное количество параллельных процессов для индексирования.



1.5.1.5 Дополнительно: настройка буферизации данных

В узле **searchindex** в файле **CommandConfig.xml** при помощи параметра **commitThreshold** можно задать, после какого количества объектов задача индексирования должна сохранять данные в индекс.

Пример с двумя синхронизируемыми индексами и рекомендуемыми значениями для индексирования (4 параллельных процесса и 250 объектов)

```
<searchindex maxParallelThreads="4" commitThreshold="250">
  <index path="//pfad/zum/lucene/index1" />
  <index path="//pfad/zum/lucene/index_backup" />
</searchindex>
```

1.5.2 Дополнительно: интеграция LDAP

Порядок выполнения операций

Настройка

Внесите в файл **\$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes\CommandConfig.xml** следующие изменения:

- Скопируйте запись **<ldapserver>** из комментария и добавьте после нее новую строку.
- Для каждого сервера LDAP необходимо указать отдельный тег **<server>**. Таким образом можно настроить несколько серверов LDAP; в этом случае будут опрашиваться все серверы LDAP до тех пор, пока не последует успешное подтверждение подлинности или не будет достигнут конец списка серверов LDAP. В зависимости от используемого протокола следующие параметры являются обязательными:

<server>:: начало настройки сервера, используемого для подтверждения подлинности LDAP.

protocol: протокол, который должен использоваться для обмена данными между сервером приложений и сервером LDAP. Допустимыми значениями являются **ldap** и **ldaps**. При значении **ldaps** обмен данными шифруется с помощью SSL.

host: имя или IP-адрес сервера LDAP, который должен использоваться.

port: порт, через который выполняется обращение к серверу LDAP.

searchUser: пользователь, используемый системой БАЗИС для контроля за введенными регистрационными данными.

searchPassword: пароль пользователя, указанного в параметре **searchUser**.

authMech: simple - данные для регистрации передаются серверу LDAP в незашифрованном виде.

Дополнительно - **keystore**: путь к файлу Keystore. Требуется только для ldaps.

Дополнительно - **keystorePassword**: пароль для указанного файла Keystore.

<serverParam>: в зависимости от указанного сервера можно использовать несколько тегов **<serverParam>**, в результате чего поиск осуществляется только в специальных узлах дерева LDAP, а не по всему дереву.

searchContext: узел в дереве LDAP, в котором будет выполняться поиск введенных регистрационных данных.

searchFilter: свойство LDAP, которое должно совпадать с введенным именем пользователя {username}.

</server>: завершение настройки сервера, используемого для подтверждения подлинности LDAP.

Пример:

```
<ldapserver>
  <server protocol="ldap" host="[DATABASE_SERVER_HOST]" port="389"
  searchUser="[LDAP_SEARCH_USER]"
  searchPassword="[LDAP_SEARCH_USER_PASSWORD]" authMech="simple"
  keystore="[KEYSTORE_FILE_PATH]"
  keystorePassword="[KEYSTORE_PASSWORD]"
  PasswordEscape="false">
    <serverparam searchContext="[LDAP_SEARCH_CONTEXT]"
    searchFilter="sAMAccountName={username}"/>
  </server>
  ..
```

Для того чтобы активировать подтверждение подлинности через LDAP, необходимо изменить параметр **SYS_USE_LDAP_LOGIN** в таблице **STFCFG_SETTING** в схеме системы БАЗИС следующим образом:

- Войдите посредством SQL*Plus в базу данных под пользователем **command**.
- Выполните следующую команду и подтвердите ее, выполнив **commit**.

```
update stfcfg_setting set value='TRUE'
where name='SYS_USE_LDAP_LOGIN';
```

- Активируйте подтверждение подлинности LDAP в разделе администрирования пользователей системы БАЗИС в модуле **Управление доступом**.



Примечание / рекомендация

Имя пользователя в системе БАЗИС должно **в точности** соответствовать имени пользователя LDAP. Особое внимание должно уделяться написанию с заглавной и строчной буквы.

Вы также можете активировать подтверждение подлинности LDAP для всех пользователей базы данных при помощи следующей команды:

```
update stcsys_user set authorization_mode='3';
```

- Завершите транзакцию, выполнив **commit**.

Если необходимо, чтобы в системе БАЗИС могли регистрироваться также пользователи, существующие только в LDAP, то для этого в системе БАЗИС можно создать запасного пользователя (fallback user) и присвоить его любой из групп, например, **role_read**. Данного пользователя можно задать в файле **CommandConfig.xml** в свойстве **fallbackUser**. Пользователь, существующий в LDAP, но не в системе БАЗИС, будет зарегистрирован в системе БАЗИС в качестве пользователя, заданного в свойстве **fallbackUser**.

Пример:

```
<property name="fallbackUser" value="[FALLBACK_USER]"/>
```



1.5.3 Дополнительно: активирование API

Генерирование API системы БАЗИС можно активировать при запуске Tomcat в файле **\$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes\CommandConfig.xml** следующим образом:

```
<property name="apiGenerationEnableAutoGeneration" value="TRUE" />
```

1.5.4 Дополнительно: шифрование файла CommandConfig.xml

Вы можете сохранить файл **CommandConfig.xml** на Вашей системе в зашифрованном виде. Для этого настроенный файл зашифровывается посредством шифрования AES (Advanced Encryption Standard) и автоматически генерируемого ключа. После этого ключ зашифровывается при помощи заданного открытого ключа RSA и добавляется к заново генерируемому файлу **CommandConfig.xmls**.

Для шифрования существующего файла **CommandConfig.xml** необходимо вызвать вспомогательную программу, используя настроенный скрипта запуска.

Настройка

Скрипт, который необходимо настроить, находится в директории **\$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes**. В зависимости от используемой операционной системы скрипт **scrambleCommandConfig.sh** (Linux/Unix) или **scrambleCommandConfig.bat** (Windows) необходимо отредактировать следующим образом:

- **TOMCAT_PATH=<Tomcat-path>** => здесь необходимо указать путь до директории Tomcat.
- **COMMAND_CLASSES** => если приложение БАЗИС находится в директории, отличной от директории webapps сервера Tomcat, то здесь необходимо ввести путь до директории **command/axis/WEB-INF/classes**.
- **COMMAND_LIB** => как и в случае с **COMMAND_CLASSES**, здесь необходимо указать путь до директории **command/axis/WEB-INF/lib**, если приложение БАЗИС находится не в директории webapps.
- **JAVA_PATH** => здесь необходимо указать путь до директории bin установленной среды Java.

Шифрование

После того как скрипт был настроен для текущей операционной системы, его можно вызвать из консоли. После исполнения скрипта в директории **command/axis/WEB-INF/classes** создается файл **CommandConfig.xmls**. Этот файл (при его наличии) загружается приложением БАЗИС при запуске. После создания и тестирования зашифрованного файла из исходного файла **CommandConfig.xml** можно удалить важные данные.

1.5.5 Дополнительно: шифрование паролей в базе данных

При помощи специальной функции можно зашифровать пароли в базе данных. Это требуется в том случае, если пароль должен проверяться не только на правильность (как, например, при регистрации в системе), но и когда он используется для регистрации в сторонней системе в незашифрованном виде.

Все известные места, для которых пароль требуется в незашифрованном виде, были подготовлены соответствующим образом. К ним относятся:

- Соединения баз данных с другими системами из модуля «Отчетность».
Таблица: STCCFG_DATABASE_CONNECTION

- Соединения для модуля «Телефонные системы».
Таблица: STCTSI_TC_CONNECTION
- Соединения баз данных из модуля «ЦОД».
Таблица: STFDCE_CONNECTION_DATABASE
- Соединения FTP из модуля «ЦОД».
Таблица: STFDCE_CONNECTION_FTP

В данных таблицах зашифровываются имя пользователя и пароль.

Для шифрования требуется пароль. Он задается для каждого из сценариев использования в файле CommandConfig.xml. Внутренний доступ осуществляется через псевдоним. В настоящее время он совпадает с именем таблицы, содержащей шифруемые столбцы.

Так как псевдоним, используемый для шифрования, жестко прописан в коде, пользователь не может его изменить.

В файл CommandConfig.xml был добавлен новый узел `<secrets>...</secrets>`. Он содержит для каждого из требуемых псевдонимов дополнительный узел `<secrets ... />`. Внутри узла предварительно настраиваются следующие три атрибута:

- alias
Псевдоним, для которого необходимо установить пароль. В настоящее время он совпадает с именем таблицы, содержащей шифруемые столбцы.
- secret
Пароль, который должен использоваться для шифрования.
- oldSecret
Пароль, который использовался для шифрования ранее. Он требуется при изменении паролей.

Использования шифрования таблицы:

Пример 1: Необходимо зашифровать таблицу

- Введите пароль в атрибут «secret».
- Перезапустите сервер приложений.

Пример 2: Изменение пароля для шифрования

- Введите предыдущий пароль в атрибут «oldSecret».
- Введите новый пароль в атрибут «secret».
- Перезапустите сервер приложений.
- При первом перезапуске пароли расшифровываются при помощи старого пароля, зашифровываются при помощи нового пароля и обновляются в базе данных. После обновления записей в базе данных oldSecret можно удалить из файла CommandConfig.xml.

Пример 3: Необходимо расшифровать таблицу (значения снова будут находиться в базе данных в незашифрованном виде)

- Процедура аналогична изменению пароля.
- secret остается пустым, а oldSecret содержит пароль, при помощи которого в текущий момент зашифровано содержимое таблицы.
- После перезапуска значения снова сохраняются в базе данных в незашифрованном виде.



- После завершения расшифровки значения `secret` и `oldSecret` можно удалить (как в состоянии при поставке).

1.5.6 Дополнительно: каталоги сообщений для других языков

В комплекте поставки могут находиться дополнительные каталоги сообщений для приложения БАЗИС (*msg_catalogs**). Их можно установить следующим образом. Перед установкой ознакомьтесь с входящим в пакет поставки файлом `readme*.txt`.

Порядок выполнения операций

- Откройте консоль и установите для параметра **NLS_LANG** значение **al32utf8**.

```
set/export NLS_LANG=[NLS_LANGUAGE]_[NLS_TERRITORY].al32utf8
```

[NLS_LANGUAGE]_[NLS_TERRITORY] может быть, например, `german_germany` или `american_america`.

- Запустите SQL*Plus и установите соединение со схемой БАЗИС.

```
sqlplus command@<TNS-имя базы данных>
```

- Исполните в SQL*Plus скрипт `update_db.sql`.

```
@update_db.sql
```

- Данные журнала сохраняются в лог-файле **update_db.log**.

Более подробную информацию о наборах символов смотрите в разделе **Ошибка! Источник ссылки не найден..**

1.5.7 Дополнительно: вызов локального клиента Putty для Telnet

В модуле «IP-адресация» в меню «Открыть в» для IPv4-адреса доступна опция вызова адреса через Telnet.

Так как в операционных системах Windows 7 и 8 вызов Telnet по умолчанию деактивирован, в качестве альтернативы можно использовать клиент **Putty**. Для этого необходимо добавить специальную запись в следующую таблицу:

- STFCFG_EXECUTE_LOCAL

Настройка Telnet (выполняется пользователем)

Для работы Putty на локальном компьютере в колонке **COMMAND** в таблице **STFCFG_EXECUTE_LOCAL** необходимо указать путь к директории установки Putty (например, `C:\Program\Putty\PUTTY.exe`).

Данная запись необходима и в том случае, если путь для каждого из пользователей различается. В этом случае после вызова Telnet открывается окно, в котором пользователь указывает свой индивидуальный путь к Putty.



Пример SQL

Приводимые ниже запросы SQL демонстрируют, как можно создать новую запись:

```
insert into STFCFG_EXECUTE_LOCAL
(ID, TYPE, COMMAND, OS, USER_ELID, GROUP_ELID, MAN_ID)
values
('TELNET', 'OS', 'putty -telnet {[hostname]}', 'DEFAULT',
'DEFAULT__USER', 'DEFAULT__GROUP', 0)
/
commit
/
```

или

```
insert into STFCFG_EXECUTE_LOCAL
(ID, TYPE, COMMAND, OS, USER_ELID, GROUP_ELID, MAN_ID)
values
('TELNET', 'OS', 'C:\Programme\Putty\PUTTY.exe -telnet
{[hostname]}', 'DEFAULT', 'DEFAULT__USER', 'DEFAULT__GROUP', 0)
/
commit
/
```

Кроме того, для различных доменов и групп можно указать свой собственный путь.

- Если для **доменов** должны использоваться различные пути, в столбце **MAN_ID** вместо значения **0** необходимо указать **идентификатор** соответствующего **домена**.
- Если для **групп** должны использоваться различные пути, в столбце **GROUP_ELID** вместо значения **DEFAULT__GROUP** необходимо указать **ELID** (уникальный идентификатор) группы пользователей.

1.5.8 Дополнительно: изменение языка для страницы входа в систему и языка пользователя

Изменение языка для страницы входа в систему можно выполнить при помощи следующего запроса SQL:

```
update stfcfg_setting set value = '<lang>'
where name = 'MOD_LANGUAGE' and user_elid = 'DEFAULT__USER'
/
commit
/
```

Если необходимо изменить язык для всех существующих пользователей, необходимо выполнить следующий запрос SQL:

```
update stfcfg_setting set value = '<lang>'
where name = 'MOD_LANGUAGE'
/
commit
/
```

Для параметра <lang> можно установить следующие значения:

- de_DE
- en_US



- ru_RU (устанавливается дополнительно)
- fr_FR (устанавливается дополнительно)

1.5.9 Дополнительно: настройка другого контекста

Для функционирования **системы БАЗИС** в другом контексте необходимо добавить (или изменить) параметр **contextRoot** в разделе **<common>** в файле **CommandConfig.xml**. Например, если **система БАЗИС** используется в контексте **/app/command**, то запись будет выглядеть следующим образом:

```
<property name="contextRoot" value="/app/command" />
```

Кроме того, также необходимо изменить контекст на сервере приложений (смотрите 1.2.3).

Так как по умолчанию подразумевается установка в корневом контексте, дополнительно необходимо соответствующим образом изменить файл index.html в директории html.

1.5.10 Дополнительно: функционирование приложения БАЗИС со шлюзом SSO (системой единого входа)

Если **система БАЗИС** используется **со шлюзом системы единого входа (СДИ Single Sign On)**, то необходимо настроить сервер приложений и приложение БАЗИС для другого контекста. Для этого необходимо выполнить шаги, описываемые в разделах 1.2.3 и 1.5.9. В качестве контекста следует использовать **/app/command**.

Кроме этого, необходимо указать JWT secret в базе данных. Для этого необходимо войти в базу данных как пользователь **command** и исполнить следующую команду:

```
begin
  spascm_content.insertCfgSetting(
    styscm_attrib_insert_tab(
      styscm_attrib_insert('NAME', 'JWT_SECRET'),
      styscm_attrib_insert('MAN_ID', '0'),
      styscm_attrib_insert('USER_ELID',
        'DEFAULT__USER'),
      styscm_attrib_insert('GROUP_ELID',
        'DEFAULT__GROUP'),
      styscm_attrib_insert('VALUE',
        '<value_of_secret>'),
      styscm_attrib_insert('DEFAULT_VALUE', ''),
      styscm_attrib_insert('DESCRIPTION', 'JWT
secret'),
      styscm_attrib_insert('SW_UNIT', 'Login'),
      styscm_attrib_insert('CATEGORY', 'SYSTEM'),
      styscm_attrib_insert('CATALOG_NAME', ''),
      styscm_attrib_insert('MSG_ID', ''),
      styscm_attrib_insert('TYPE', 'STRING'),
      styscm_attrib_insert('ALLOW_EMPTY_VALUE', 'N'),
      styscm_attrib_insert('ENCRYPTION_STATUS',
        'SHOULD_BE_ENCRYPTED'),
      styscm_attrib_insert('CUSTOM', 'N')));
end;
/
commit
/
```

2 Дополнительные варианты установки

2.1 Вариант 2

В данном варианте к одной общей базе данных параллельно подсоединено несколько серверов Tomcat. Этот вариант можно, например, использовать для равномерного распределения нагрузки или повышения отказоустойчивости веб-серверов.

2.1.1 Настройки для мультисерверной архитектуры (кластера)

Для того чтобы клиентский сеанс был действительным на всех серверах приложений, систему БАЗИС необходимо настроить специальным способом. Кроме того, в таком варианте все динамически создаваемое содержимое (например, графические изображения для заново импортированных цифровых моделей) должно автоматически распределяться по всем серверам приложений.

Настройка

Внесите в файл `$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes\CommandConfig.xml` следующие изменения:

Опцию **multiserver** на всех серверах приложений необходимо установить в **true**.

```
<!--specifies whether the multiserver feature is enabled ("true") or disabled ("false")-->  
<property name="multiserver" value="true"/>
```

Опцию **serverJobs** на одном из серверов приложений необходимо установить в **enabled**, а на всех остальных - в **disabled**.

```
<!-- specifies whether the server can create threads or not and if jobs should be  
scheduled. With multiserver: only one server should create threads! disabled -  
ThreadManager and jobs disabled, enabled - ThreadManager and jobs enabled,  
threadManagerOnly - ThreadManager enabled and jobs disabled, cronjobsOnly -  
ThreadManager disabled and jobs enabled-->  
<property name="serverJobs" value="enabled"/>
```

Поиск

При функционировании в мультисерверном режиме индекс поиска для экземпляра приложения БАЗИС, для которого должно выполняться индексирование, должен храниться на совместно используемом ресурсе; при этом индекс поиска необходимо сделать доступным для всех других серверов



Примечание / рекомендация

В описываемых далее вариантах установки также необходимо выполнить мультисерверные настройки; на них мы отдельно останавливаться больше не будем.

2.2 Вариант 3

В данном варианте сервер Apache работает в связке с Tomcat; при этом сервер Apache обращается к базе данных через экземпляр БАЗИС сервера Tomcat. В этом варианте основной поток данных проходит через Apache. Сервер Apache обращается только к экземпляру приложения БАЗИС сервера Tomcat и формирует страницу посредством Tomcat.



2.2.1 Настройка сервера Apache

В данном варианте необходимо учитывать, что сервер Apache является лишь веб-сервером, а не сервером приложений как Tomcat. Поэтому между сервером Apache и сервером Tomcat устанавливается соединение, чтобы сделать возможным формирование страниц посредством сервера Apache.

Для установления соединения между серверами Apache и Tomcat необходимо убедиться в том, что при установке Apache были установлены модули **mod_proxy** и **mod_proxy_ajp** и что эти модули были активированы в конфигурации сервера Apache.

При стандартной установке сервера Apache в 64-разрядной ОС Suse Linux модули находятся в директории **usr/lib64/apache2-prefork/**. Чтобы они загружались при запуске Apache, модули прописываются в файле **loadmodule.conf** в каталоге **/etc/apache2/sysconfig.d/**. Если запись в файле **loadmodule.conf** не работает, модули необходимо активировать при помощи команды **a2enmod**, и они будут загружены при следующем запуске. В других дистрибутивах файл конфигурации и место хранения модулей могут отличаться.

Если необходимые модули активированы, осталось только соответствующим образом настроить запись **vhosts** сервера Apache. Запись vhosts различается в зависимости от того, используется SSL или нет.

В примере для веб-сервера Apache используется порт 80.

Без использования SSL (HTTP)

Ниже представлена запись vhosts, когда сертификат SSL / шифрование SSL не используется.

```
<VirtualHost *:80>
    ServerAdmin info@sdisoft.ru
    ServerName sample.sdisoft.ru
    ErrorLog /var/log/apache2/sample.sdisoft.ru-error_log
    CustomLog /var/log/apache2/sample.sdisoft.ru-access_log combined
    HostnameLookups Off
    UseCanonicalName Off
    ServerSignature On

# Apache <> Tomcat AJP-коннектор (доступно из диапазона IP-адресов
192.168.x.x)
    <Proxy *>
        AddDefaultCharset Off
        Order deny,allow
        Allow from all
        Allow from 192.168
    </Proxy>

    ProxyPass / ajp://localhost:8009/
    ProxyPassReverse / ajp://localhost:8009/
# /Apache <> Tomcat AJP-коннектор

</VirtualHost>
```



С использованием SSL (HTTPS)

Если настройка виртуальных хостов должна выполняться посредством шифрования SSL с использованием сертификата SSL, конфигурацию необходимо изменить следующим образом.

```
<IfDefine !NOSSL>
<VirtualHost _default_:443>
    ServerAdmin sample@sampledom.ru
    ServerName sample. sampledom.ru
    ErrorLog /var/log/apache2/sample. sampledom.ru-error_log
    TransferLog /var/log/apache2/sample. sampledom.ru-access_log
    SSLEngine on
    SSLCipherSuite
ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL
    SSLCertificateFile /etc/apache2/ssl.crt/sample. sampledom.ru.crt
    SSLCertificateKeyFile /etc/apache2/ssl.key/sample.
sampledom.ru.key
    <Files ~ "\.(cgi|shtml|phtml|php3?)$" >
        SSLOptions +StdEnvVars
    </Files>
    <Directory "/srv/www/cgi-bin">
        SSLOptions +StdEnvVars
    </Directory>
    SetEnvIf User-Agent ".*MSIE.*" \
        nokeepalive ssl-unclean-shutdown \
        downgrade-1.0 force-response-1.0
    CustomLog /var/log/apache2/ssl_request_log ssl_combined
# Apache <> Tomcat AJP-коннектор (доступно из диапазона IP-адресов
192.168.x.x)
    <Proxy * >
        AddDefaultCharset Off
        Order deny,allow
        Allow from all
        Allow from 192.168
    </Proxy>
    ProxyPass / ajp://localhost:8009/
    ProxyPassReverse / ajp://localhost:8009/
# /Apache <> Tomcat AJP-коннектор
</VirtualHost>
</IfDefine>
```

Предпосылкой для настройки SSL является наличие файлов SSL Certificate и SSL Certificate Key.

Настройка порта AJP-коннектора в Tomcat

Вы найдете порт AJP для параметров *ProxyPass* и *ProxyPassReverse* в файле Server.xml на соответствующем сервере Tomcat.

```
<!-- Define an AJP 1.3 Connector on port 8009 -->
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
```

2.2.2 Дополнительно: распределение приложения БАЗИС между Tomcat и Apache

В данном варианте приложение БАЗИС подразделяется на динамическую и статическую составляющую. Динамическая составляющая (директория **axis** и **wmicons**) работает на сервере Tomcat, а статическая составляющая (остальное содержимое директории **command**) исполняется на сервере Apache.

В данном случае исходят из того, что шаги настройки для варианта 3 уже были выполнены. Содержимое приложения БАЗИС сохраняется как на сервере Apache, так и на сервере Tomcat в поддиректории **command**.

Настройка сервера Apache

Для того чтобы подготовить сервер Apache для работы со статической составляющей и одновременно подготовить соединение с Tomcat для работы с динамической составляющей, необходимо выполнить следующие операции.

- Сначала необходимо настроить сервер Apache таким образом, чтобы он получал статическую составляющую приложения БАЗИС из своей собственной директории **htdocs**. Для этого необходимо добавить в запись **DocumentRoot** в файле **httpd.conf** данные о директории **/command**.

```
DocumentRoot "/apache/apache242command/htdocs/command"
<Directory "/apache/apache242command/htdocs/command">
```

- Для ранее настроенного vHost необходимо изменить параметры **ProxyPass** и **ProxyPassReverse** следующим образом:

```
ProxyPass /axis ajp://localhost:8009/axis
ProxyPassReverse /axis http://localhost:80/axis

ProxyPass /wmicons ajp://localhost:8009/wmicons
ProxyPassReverse /wmicons http://localhost:80/wmicons
```



Примечание / рекомендация

В целях безопасности для **ProxyRequests** должно быть установлено значение **Off**. Также необходимо активировать модуль **mod_proxy_http** в файле **httpd.conf**.

- ✓ Теперь необходимо перезапустить сервер Apache.

Настройка сервера Tomcat

- На сервере Tomcat в файле **server.xml** в директории **\$TOMCATHOME/conf/** переменные контекстного пути для директорий **axis** и **wmicons** необходимо настроить следующим образом:

```
<Context path="/axis" docBase="command/axis" debug="0"
reloadable="false" crossContext="true" swallowOutput="true"/>
<Context path="/wmicons" docBase="command/wmicons" debug="0"
reloadable="false" crossContext="true" swallowOutput="true"/>
```

- Запись контекстного пути «/» из файла **server.conf** можно удалить.

2.3 Вариант 4

В данном варианте несколько установленных серверов Apache и Tomcat используются на разном аппаратном обеспечении. Нагрузка распределяется посредством распределителя нагрузки (Load Balancer) по различным физическим серверам, в результате чего для всех пользователей обеспечивается оптимальная производительность. Все работающие параллельно веб-серверы обращаются при этом к одной и той же базе данных.

Для того чтобы обеспечить бесперебойный, мультисерверный режим работы с распределителем нагрузки, необходимо активировать функцию распределителя нагрузки **BALANCER_SESSION_STICKY**. В противном случае могут возникнуть проблемы с работой некоторых функций, например, с импортом данных объектов.

2.3.1 Настройка распределителя нагрузки (Load Balancer)

Для работы с распределителем нагрузки необходимо убедиться в том, что на сервере Apache установлен модуль **mod_proxy_balancer** и что он загружается при запуске сервера Apache.

Чтобы использовать основные функции распределителя нагрузки, необходимо добавить в файл **httpd-vhosts.conf** на сервере Apache, который отвечает за распределение нагрузки, следующую запись:

```
<Proxy balancer://projekt1>
BalancerMember http://192.168.0.30:80
BalancerMember http://192.168.0.31:80
</Proxy>
<Location>
ProxyPass / balancer://projekt1/
</Location>
```

IP-адрес и порт можно изменить в соответствии с портами серверов Apache, между которыми должна распределяться нагрузка.



Примечание / рекомендация

Так как в конфигурации серверов Apache для формирования страниц был ограничен IP-адрес для доступа, то сервер должен находиться в том же заданном диапазоне IP-адресов, что указан в записи, или необходимо изменить следующие строки в файлах **httpd-vhosts.conf** на серверах Apache.

```
<Proxy *>
  AddDefaultCharset Off
  Order deny,allow
  Allow from all
  Allow from 192.168      <= Данную строку необходимо изменить /
удалить.
</Proxy>
```



Примечание / рекомендация

Если строка **Allow from xxx.xxx** удаляется полностью, вызов можно выполнить с любого IP-адреса. Это может быть небезопасно, поэтому рекомендуется изменить диапазон IP-адресов 192.168 в соответствии с диапазоном IP-адресов распределителя нагрузки.

2.3.2 Настройка функции BALANCER_SESSION_STICKY



Примечание / рекомендация

Данная конфигурация протестирована только для Apache версии 2.0 и старше. У более ранних версий или других поставщиков Apache данная конфигурация может отличаться, или данная функция может полностью отсутствовать / иметь другое название. В этом случае для правильной настройки необходимо обратиться к соответствующей документации сервера приложений.

Как уже было описано в начале данного варианта, функцию необходимо активировать, чтобы приложение БАЗИС работало без ошибок. Это связано с тем, что, если пользователь запускает выгрузку данных, которая успешно завершается, при повторном установлении соединения пользователь может попасть на второй веб-сервер. Но так как файл туда загружен не был, то у пользователя не будет доступа к выгруженным данным. Функция Session Sticky гарантирует, что каждый пользователь окажется на том же самом веб-сервере вплоть до завершения сеанса работы.

Для этого необходимо изменить конфигурацию в файле **httpd-vhosts.conf** сервера Apache, отвечающего за распределение нагрузки, следующим образом:

```
<Proxy balancer://projekt1>
BalancerMember http://192.168.0.30:80 route=19216803080
BalancerMember http://192.168.0.31:80 route=19216803180
</Proxy>
<Location>
ProxyPass balancer://projekt1/lbmethod=byrequests stickysession=JSESSIONID
ProxyPassReverse http://192.168.0.30:80/
ProxyPassReverse http://192.168.0.31:80/
</Location>
```

Для параметра **lbmethod** на данный момент доступно 3 различных значения:

1. **Byrequest:** распределяет нагрузку в зависимости от количества запросов на серверах.

При помощи параметра **loadfactor** (значение от 0 до 100) для BalancerMember можно задать, какой процент нагрузки будет распределяться на какие серверы.

```
BalancerMember http://192.168.0.30:80 loadfactor=4 route=19216803080
BalancerMember http://192.168.0.31:80 loadfactor=6 route=19216803180
```

Первый элемент (member) получит 40% нагрузки, а второй - 60%.

2. **Bytraffic:** распределяет нагрузку в зависимости от трафика на серверах.
3. **Bybusyness:** нагружает тот сервер, который меньше всего задействован (работает в **холостом режиме**).

3 Настройка дополнительных модулей

3.1 Инициализация метасхемы БАЗИС

Условия

Для метасхемы БАЗИС требуется пользователь базы данных, называемый в дальнейшем **cmd_meta** (имя может быть произвольным). Чтобы создать метасхему, пользователь **cmd_meta** должен обладать правами на схему пользователя **command**.

Порядок выполнения операций

Установка

- Создайте пользователя **cmd_meta**:

```
create user cmd_meta identified by cmd_meta;
grant alter session to cmd_meta;
grant create session to cmd_meta;
grant create view to cmd_meta;
grant create synonym to cmd_meta;
grant execute on command.spamsa_make to cmd_meta;
```

- Войдите в базу данных под пользователем **command** и подготовьте схему БАЗИС для метасхемы:

```
exec spamsa_metaschema.MetaViewForUser('cmd_meta');
```

- Войдите в базу данных под пользователем **cmd_meta** и создайте метасхему:

```
exec
command.spamsa_make.createSynonyms('cmd_meta','command');
```

- Войдите в базу данных под пользователем **command** и задайте для модуля **Отчетность** (редактор запросов) подключение к базе данных по умолчанию:

```
update STCCFG_DATABASE_CONNECTION
set server = '<БАЗИС_ДБ-Сервер>', instance = '<БАЗИС_ДБ-
Экземпляр>', port = '<Слушающий_Порт>', login = 'cmd_meta',
password = 'cmd_meta'
where ID = 'DEFAULT' and FIX = 2;
```

3.2 Почтовый сервер по умолчанию

Условия

В различных модулях системы БАЗИС (например, «Отчетность») можно настроить, чтобы система БАЗИС отправляла электронные сообщения на заданный почтовый сервер. Для этого необходимо настроить почтовый сервер по умолчанию.

Порядок выполнения операций

Настройка

Внесите в файл `$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes\CommandConfig.xml` следующие изменения:

- Почтовый сервер, через который система БАЗИС отправляет электронные сообщения:

```
<property name="mailServerName" value="smtp.localhost.de"/>
```

- Порт, по которому доступен почтовый сервер:

```
<property name="mailServerPort" value="25"/>
```

- Настройка, ожидает ли почтовый сервер подтверждение подлинности при отправке электронных сообщений (true / false):

```
<property name="mailServerUseAuth" value="false"/>
```

- Пользователь и пароль для подтверждения подлинности на почтовом сервере, если для `mailServerUseAuth` установлено значение `true`:

```
<property name="mailServerUsername" value=""/>
```

```
<property name="mailServerPassword" value=""/>
```

- Отправитель электронных сообщений, посылаемых системой БАЗИС, если не указан никакой другой отправитель:

```
<property name="mailSenderAddress" value="command@fnt.de"/>
```

- Адрес, на который могут отправляться ответы от почтового сервера при возникновении проблем (например, если неизвестен адрес получателя):

```
<property name="mailErrorReplyAddress" value="command@fnt.de"/>
```

- Настройка, будет ли система БАЗИС сохранять электронное сообщение в базе данных после его отправки:

```
<property name="saveMailInDB" value="false"/>
```

3.3 Редакторы запросов

Условия

В редакторе запросов системы БАЗИС можно выполнять отчеты в качестве задач. Для этого необходимо настроить вход в систему БАЗИС (пользователь / группа / домен). Такие задания могут экспортировать отчеты в файлы различных форматов и отправлять их в виде электронных сообщений определенному списку получателей. Для отправки электронной почты используется почтовый сервер по умолчанию, описанный выше. Если необходимо, чтобы электронные сообщения автоматически отправлялись из редактора запросов, то также необходимо настроить почтовый сервер по умолчанию, как описано в разделе *ПОЧТОВЫЙ сервер по умолчанию*.

Порядок выполнения операций

Настройка

Внесите в файл `$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes\CommandConfig.xml` следующие изменения:

- Пользователь, под которым задача редактора запросов регистрируется в системе БАЗИС:

```
<property name="qedJobUserName" value="command"/>
```

- Роль, под которой должны выполняться задачи редактора запросов:

```
<property name="qedJobUserGroupId" value="role_admin_1001"/>
```

- Домен, под которым должны выполняться задачи редактора запросов:

```
<property name="qedJobManId" value="1001"/>
```

- Если пользователь назначен указанной группе или указанному домену (по умолчанию - группе):

```
<property name="qedJobUGType" value="G"/>
```

- Электронный адрес отправителя для сообщений, посылаемых редактором запросов:

```
<property name="qedMailSenderAddress" value="command.query_editor@fnt.de"/>
```

- Для того чтобы отчеты редактора запросов можно было запускать в качестве задач, их необходимо активировать:

```
<property name="serverJobs" value="enabled"/>
```



Примечание / рекомендация

В мультисерверном режиме активацию следует настраивать только на одном из серверов.

3.4 Радиосети

Условия

В модуле «Радиосети» отправка электронных сообщений может осуществляться во время циклических проверок или после действий, инициированных в ручном режиме. Для этого помимо почтового сервера по умолчанию необходимо настроить несколько дополнительных параметров.

Порядок выполнения операций

Настройка

Внесите в файл `$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes\CommandConfig.xml` следующие изменения:

- Адресат, который получает копии всех уведомлений о новых / измененных объектах в модуле «Радиосети»:

```
<property name="RNM_[NEW|CHANGED]_OBJECT_CC_EMAIL_ADDRESS"
value="sdi.rnm@sdisoft.ru"/>
```

- Адресат, который получает копии всех уведомлений об объектах модуля «Радиосети» с истекшим сроком действия:

```
<property name="RNM_SEND_VALIDITY_MAIL" value="TRUE"/>
```

- Отправлять электронные сообщения для новых или измененных объектов модуля «Радиосети»?

```
<property name="RNM_VALIDITY_CC_EMAIL_ADDRESS"
value="sdi.rnm@sdisoft.ru"/>
```

- Отправлять электронные сообщения для объектов модуля «Радиосети» с истекшим сроком действия?

```
<property name="RNM_SEND_MAIL_ON_CREATE_OR_CHANGE"
value="TRUE"/>
```

Кроме того, в файле `$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes\radiomgmt.properties` необходимо изменить следующие параметры:

- Промежуток времени (в секундах), в течение которого подпроцесс «Радиосети» проверяет необходимость своего очередного выполнения:
THREAD_SLEEP 300
- Время для исполнения подпроцесса (в данном случае 5 часов утра):
THREAD_HOUR 5
- Параметр `THREAD_CYCLE` нельзя изменить:
THREAD_CYCLE 3600
- Пользователь системы БАЗИС, от имени которого будет исполняться подпроцесс:
THREAD_USER command

- Пароль, который будет использоваться пользователем подпроцесса для входа в систему БАЗИС.
THREAD_PASS command
- Домен, под которым пользователь будет регистрироваться в системе:
THREAD_MAND_ID 1001
- Имя группы, под которой пользователь будет регистрироваться в системе:
THREAD_MAND_GROUP_ID role_admin_1001
- Задаёт, назначен ли пользователь непосредственно домену (U) или группе (G):
THREAD_MAN_UG_TYPE G

3.5 Управление ЦОД

3.5.1 Активирование модулей для управления ЦОД

Для активирования управления ЦОД сначала необходимо остановить Tomcat, а затем в файле `$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes\CommandConfig.xml` удалить знаки комментариев для следующих записей:

```
<module moduleBaseClass="de.fnt.command.biz.modules.powermgmt.PowerMgmt"
name="powermgmt"/>
<module initClass="de.fnt.command.biz.modules.datacentercockpit.DataCenterCockpit"
moduleBaseClass="de.fnt.command.biz.modules.datacentercockpit.DataCenterCockpit"
name="datacentercockpit"/>
<module initClass="de.fnt.command.biz.modules.aircon.Aircon" name="aircon"/>
<module
initClass="de.fnt.command.biz.modules.footprint3d.Footprint3D"
moduleBaseClass="de.fnt.command.biz.modules.footprint3d.Footprint3D"
name="footprint_3d"/>
```

Кроме того, модули должны быть активированы в таблице `stfsys_sw_unit`. Выполните для этого в схеме системы БАЗИС следующую команду SQL и подтвердите ее, выполнив `«commit;»`:

```
update stfsys_sw_unit set enabled = 'Y'
where sw_unit in
('AIRCON', 'DATACENTERCOCKPIT', 'DATACENTERCOCKPIT:FOOTPRINT',
'POWERMGMT', 'POWERMGMT:CIRCUITDIAGRAMLIST', 'POWERMGMT:GRAPHICAL',
'FOOTPRINT_3D');
```

После этого запустите Tomcat снова.



Примечание / рекомендация

Активирование функциональности для управления ЦОД следует выполнять только после консультации с компанией СДИ. Это означает, что модуль не должен активироваться / деактивироваться по собственному желанию. Кроме того, требуется, чтобы компания СДИ активировала данные управления ЦОД в файле DAT в библиотеке цифровых моделей.

3.5.2 Установка

- В модуле «Электропитание» в меню **Импорт** -> **Запуск импорта** можно вручную запустить импорт данных измерения. Какой именно файл будет считываться в ходе импорта, задается посредством экземпляра базы данных в таблице **STFCFG_JOB_PARAM**. Путь к файлу можно настроить в схеме БАЗИС при помощи следующей команды SQL (а затем подтвердить, выполнив `«commit;»`):



```
update stfcfg_job_param t
  set value = '<абсолютный_путь_с_именем_файла>'
where job_elid =
  (select infos from stccfg_job t
  where job_name = 'Import Measuring Data' and job_group =
  'Powermgmt') and name = 'IMPORT_FILE';
```

- Если необходимо, чтобы импорт данных измерений выполнялся в качестве задачи, то в схеме БАЗИС в таблице **STCCFG_JOB** при помощи следующей команды SQL необходимо активировать задачу **Import Measuring Data** из группы задач **Powermgmt** (а затем подтвердить, выполнив «**commit**;»):

```
update STCCFG_JOB set
  user_name = '<имя_пользователя>',
  man_id = '<ID_домена>',
  user_group_id = '<имя_пользователя_группы>',
  ug_type = '<тип_пользователя_группы>'
where job_name = 'Import Measuring Data' and job_group =
  'Powermgmt';
```

- **имя_пользователя:** Пользователь, от имени которого выполняется задание.
- **ID_домена:** Домен, под которым пользователь регистрируется в системе.
- **имя_пользователя_группы:** Имя пользователя или группы, под которым / которой пользователь регистрируется в домене.
- **тип_пользователя_группы:** Задаёт, означает ли значение в параметре **user_group_id** группу (G) или пользователя (U).
- Кроме того, в интерфейсе модуля «Электропитание» (**Импорт** → **Конфигурация**) необходимо задать время выполнения задачи. Опция **Импорт автоматически активный** должна быть активирована, чтобы задача выполнялась автоматически.
- Для выполнения настроенных задач в системе БАЗИС необходимо установить значение **enabled** для параметра **serverJobs** в файле **\$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes\CommandConfig.xml**.

```
<property name="serverJobs" value="true"/>
```



Примечание / рекомендация

В мультисерверном режиме активацию следует настраивать только на одном из серверов.



Переключение с БТУ/ч на кВт

В управлении ЦОД можно изменить используемые единицы измерения с БТУ/ч на кВт. Для этого в схеме БАЗИС необходимо выполнить следующие команды SQL и подтвердить их, выполнив «**commit**»:

```
update stfcfg_setting set value = 'kW'
where name = 'SYS_CLIMATE_CAPACITY_UNIT';

update stfsys_msg_catalog set msg_text = 'kW', custom = 'Y'
where catalog_name = 'CLIMATE_CAPACITY_UNIT' and msg_id = 28;
```

3.6 Телеком

Для сетевых элементов существует зона по умолчанию, если сетевому элементу зона не назначается вручную. Если такая зона отсутствует, имя можно изменить при помощи следующей команды SQL и выполнив «**commit**»:

```
update stfcfg_setting set value='[Autonomous system]'
where name='TCO_NE_DEFAULT_ZONE_NAME;
```

Если зона уже существует, переименовать ее может только служба поддержки СДИ.

Чтобы в модуле «Телеком» корректно выполнялся импорт умляутов, необходимо соответствующим образом изменить кодировку файла Excel, который предполагается импортировать. Для этого в базе данных в таблице **STFCFG_SETTING** имеется запись **MOD_EXCEL_IMPORT_JXL_ENCODING**.

Для обновления записи в схеме БАЗИС необходимо выполнить следующую команду SQL и подтвердить при помощи «**commit**»:

```
update stfcfg_setting set value='<ENCODING>'
where name='MOD_EXCEL_IMPORT_JXL_ENCODING';
```

3.7 Наряды

В модуле «Наряды» можно активировать автоматическую задачу, которая будет отслеживать статус эскалации существующих подзадач. При изменении статуса задача отправляет электронное сообщение (смотрите раздел 3.2) адресатам, которые были указаны в настройках эскалации для данного статуса эскалации. Для выполнения задачи требуется зарегистрироваться в системе БАЗИС.

Настройка

Выполните в файле **\$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes\CommandConfig.xml** следующие изменения:

- Пользователь, от имени которого задания модуля «Наряды» выполняются в системе БАЗИС:



```
<property name="ormJobUserName" value="command"/>
```

- Пароль для пользователя, указанного выше:

```
<property name="ormJobUserPassword" value="command"/>
```

- Домен, под которым указанный пользователь регистрируется в системе БАЗИС:

```
<property name="ormJobUserMandantId" value="1001"/>
```

- Группа, от имени которой выполняются задания:

```
<property name="ormJobUserGroup" value="role_admin_1001"/>
```

3.8 3D план помещения

Модуль «3D план помещения» представляет собой графический модуль, работающий в окне браузера и использующийся для визуализации ЦОД в формате 3D. Модуль можно запустить из модуля «ЦОД», в результате чего на основе 2D-плана создается виртуальное 3D-представление.

3.8.1 Активирование модуля «3D план помещения»

- Остановите Tomcat.
- Активируйте запись в файле CommandConfig.xml (удалите знаки комментариев <!-- -->).

```
<module name="footprint_3d"
moduleBaseClass="de.fnt.command.biz.modules.footprint3d.Footprint3D"
initClass="de.fnt.command.biz.modules.footprint3d.Footprint3D" />
```

- Выполните следующую команду в SQL Plus / SQL Editor и подтвердите выполнение посредством «commit;».

```
update stfsys_sw_unit set enabled='Y'
where sw_unit = 'FOOTPRINT_3D';
```

- Скопируйте архив WAR в директорию **appBase** (путь по умолчанию для установочных файлов Tomcat - \$CATALINA_BASE/webapps). Как правило, приложение устанавливается автоматически при запуске сервера.
- Запустите Tomcat.

(идентификатор роли 75).

3.9 SOAP API

Система БАЗИС предоставляет SOAP API, в основе которых лежит технология интеграционного слоя. По умолчанию интерфейс прикладных программ (API)

деактивирован. **Интерфейс SOAP API будет удален в следующем главном релизе системы БАЗИС.** Необходимо выполнить своевременную миграцию на REST API.

Настройка

Выполните следующие изменения в файле **\$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes\CommandConfig.xml**. Оба параметра находятся в разделе «specific».

- Активирует / деактивирует интерфейс SOAP API:

```
<property name=" apiGenerationEnableAutoGeneration"
value="true"/>

<property name=" apiGenerationEnabledFor" value="SOAP"/>
```

Так как для SOAP API требуется большое количество памяти, все сервисы сущностей по умолчанию деактивированы. В зависимости от потребностей Вы можете активировать только отдельные сущности (рекомендуется) или все сущности. При активировании всех сущностей потребление памяти составляет около 0,5 ГБ. Настройка выполняется в файле **\$TOMCAT_HOME\webapps\command\axis\WEB-INF\config\apisoapservices.properties file**:

```
# активирует отдельные сущности
DeviceAll
Zone
Person
custom.Key # ключ для пользовательской сущности

# активирует все сущности (при помощи звездочки)
*

# активирует все сущности каналов «Телеком»
(подстановочный знак)
ServiceTelco*
```

3.10 Запросы на прокрутку (для интеграционного слоя)

Запросы на прокрутку используются для поточных запросов больших объемов данных посредством интеграционного слоя. При использовании запросов на прокрутку данные для кэширования сохраняются на файловой системе. Путь, по которому сохраняются данные прокрутки, можно задать в файле CommandConfig.xml. Если возможно, место сохранения необходимо выбрать на другом разделе диска или на другом диске, чтобы не зависеть от сервера системы БАЗИС. Благодаря этому можно избежать проблем в эксплуатации, если свободное место на диске закончится. Более подробная информация о необходимом размере памяти приводится в документации по системным требованиям.

Внимание: Запросы на прокрутку не поддерживаются в мультисерверном режиме с использованием распределителя нагрузки. Запросы на прокрутку должны выполняться к одному и тому же серверу приложений. Кроме того, путь к запросам на прокрутку не может использоваться несколькими серверами приложений.



Настройка

Выполните следующие изменения в файле **\$TOMCAT_HOME\webapps\command\axis\WEB-INF\classes\CommandConfig.xml**. Параметр находится в разделе «specific».

- Задает путь до данных прокрутки:

```
<property name="scrollQueryDataLocation"
value="s:/scrollQueryData/" />
```

3.11 Клиентское расширение БАЗИС

Клиентское расширение БАЗИС (КРБ) сменяет функциональность JAVA_APPLETS. КРБ - клиентское приложение, связь с которым может устанавливаться из системы БАЗИС. По умолчанию обмен данными между системой БАЗИС и КРБ осуществляется через порт HTTP 9039.

Для настройки порта необходимо изменить **MOD_COMMAND_EXTENSION_PORT** в таблице **STFCFG_SETTING** в схеме БАЗИС следующим образом:

- Войдите посредством SQL*Plus в базу данных под пользователем **command**.
- Выполните следующую команду, используя действительный номер от 1024 до 65535 вместо <порта>, и подтвердите транзакцию при помощи **commit**.

```
update stfcfg_setting set value='<PORT>'
where name='MOD_COMMAND_EXTENSION_PORT';
```

- Будет ли использоваться КРБ, можно настроить для каждого из пользователей отдельно в пользовательском интерфейсе системы БАЗИС в меню Еще -> Пользовательские настройки -> Глобальные настройки -> Общие.

4 Обновление существующих систем

4.1 Обновление программы - директория приложения БАЗИС

В большинстве случаев при серьезных изменениях системы (например, при обновлении на новую версию) обновленное приложение поставляется СДИ в виде директории, которая содержит все файлы, необходимые для работы Tomcat. Данную директорию необходимо установить на соответствующем экземпляре Tomcat следующим образом:

- Остановите экземпляр Tomcat, который необходимо обновить.
- Измените имя для поддиректории **command**, находящейся в директории **\$TOMCAT_HOME\webapps**, на **command_<ГГГГ-ММ-ДД>**.
- Распакуйте входящий в пакет поставки файл **command.zip** в директорию **\$TOMCAT_HOME\webapps**. Теперь в папке должны находиться следующие две папки:



Рисунок4-1 Папка для резервного копирования

- Если обновление базы данных осуществляется при помощи скрипта, а не дампа базы данных, необходимо скопировать файл **C6UpdateTask.properties** из директории **command/axis/WEB-INF/classes/**, находящейся в резервной копии, в новую директорию **command**. При этом файл из комплекта поставки заменяется на первоначальный.
- Измените новый файл **CommandConfig.xml** из директории **command/axis/WEB-INF/classes/** в соответствии со значениями в исходном файле.



Примечание / рекомендация

Не всегда возможно просто заменить старый файл **CommandConfig.xml** на новый. Нельзя исключать ошибок, обусловленных старой версией файла.



Примечание / рекомендация

В любом случае необходимо изменить следующие значения:

Порт Apache

```
<property name="apachePort" value="xxxx"/>
```

Пример:

```
<!--apache webserver port for protocol http -->
<property name="apachePort" value="80"/>
```

Рисунок 4-2 Пример для порта

Если используется SSL:

```
<property name="apachePortSSL" value="xxxx"/>
```

Пример:

```
<!--apache webserver port for protocol https -->
<property name="apachePortSSL" value="443"/>
```

Рисунок 4-3 Пример для порта SSL

Путь до директории wmicons

```
<property name="wmiIconsHome" value="PathToWmiconsFolder"/>
```

Пример:

```
<!--path to the wmi icons folder-->
<property name="wmiIconsHome" value="/export/daten2/apache-tomcat-5.5.26/instances/testinstanz/webapps/command/wmicons"/>
```

Рисунок 4-4 Пример для wmicons

Подключение к БД

```
<db connectionIdleTime="0" connectionValidTime="0" id="default"
instance="DbInstance" maxPoolSize="4" minPoolSize="2"
password="CommandDbPassword" username="CommandDbUsername">
  <server port="DbListenerPort" url="DbServerUrl"/>
</db>
```

Пример:

```
<dbserver>
<!--default database instance-->
<!--connectionValidTime in minutes: time a connection is valid, value 0 means unlimited -->
<!--connectionIdleTime in minutes: time a connection is valid if not in use, value 0 means unlimited -->
<db connectionIdleTime="0" connectionValidTime="0" id="default" instance="testinstanz" maxPoolSize="4" minPoolSize="2" password="command" username="command">
  <server port="1521" url="sora81"/>
</db>
```

Рисунок 4-5 Пример для базы данных

- Скопируйте следующие поддиректории из ранее заархивированной директории **command_<ГГГГ-ММ-ДД>/wmicons** в новую директорию **command/wmicons**:

- png
- png_rear
- swf
- swf_rear

Если необходимо, перезапишите существующие файлы.

- При обновлении приложения БАЗИС с версии ≤ 9.3 до $\Rightarrow 9.4$ необходимо дополнительно настроить Tomcat и установить сервер лицензий в соответствии с инструкциями в разделах *УСТАНОВКА СЕРВЕРА ПРИЛОЖЕНИЙ* и *УСТАНОВКА СЕРВЕРА ЛИЦЕНЗИЙ*.
- **После успешного обновления базы данных** снова запустите экземпляр Tomcat (смотрите раздел **Ошибка! Источник ссылки не найден.**).
- Если обновление прошло без ошибок, Вы можете удалить резервную копию ***command_<ГГГГ-ММ-ДД>***.

4.2 Обновление программы - пакеты исправлений и заплат

Незначительные изменения поставляются СДИ в виде пакетов исправлений (hoftix) и заплат (patch). В этом случае в директорию ***command*** соответствующего экземпляра Tomcat добавляются только новые версии файлов. Установка пакета исправлений / заплат выполняется следующим образом:

- Остановите экземпляр Tomcat, который необходимо обновить.
- Скопируйте поддиректорию ***command*** из директории ***\$TOMCAT_HOME\webapps*** и переименуйте резервную копию в ***command_<ГГГГ-ММ-ДД>***.



Примечание / рекомендация

Необходимо создать резервную копию, а не просто переименовать исходную директорию.



Рисунок4-6 Директория для резервного копирования

- Распакуйте входящий в пакет поставки файл ***command.zip*** в директорию ***\$TOMCAT_HOME\webapps***. Если необходимо, перезапишите существующие файлы / поддиректории в существующей директории ***command***.



Примечание / рекомендация

Так как директория ***command*** уже содержится в файле поставки ***command.zip***, всегда распаковывайте ZIP-файл непосредственно в директорию ***\$TOMCAT_HOME\webapps***.

- Если в комплекте поставки содержится файл ***CommandConfig.xml***, необходимо отредактировать значения в новом файле ***CommandConfig.xml*** в соответствии с исходным файлом.

Порт Apache

```
<property name="apachePort" value="xxxx"/>
```

Пример:

```
<!--apache webserver port for protocol http -->
<property name="apachePort" value="80"/>
```

Рисунок4-7 Пример для порта

Если используется SSL:

```
<property name="apachePortSSL" value="xxxx"/>
```

Пример:

```
<!--apache webserver port for protocol https -->
<property name="apachePortSSL" value="443"/>
```

Рисунок 4-8 Пример для порта SSL

Путь до директории wmicons

```
<property name="wmlconsHome" value="PathToWmiconsFolder"/>
```

Пример:

```
<!--path to the wm icons folder-->
<property name="wmlconsHome" value="/export/daten2/apache-tomcat-5.5.26/instances/testinstanz/webapps/command/wmicons"/>
```

Рисунок 4-9 Пример для wmicons

Подключение к БД

```
<db connectionIdleTime="0" connectionValidTime="0" id="default"
instance="[INSTANCE_NAME]" maxPoolSize="4" minPoolSize="2"
password="[DATABASE_USER_PASSWORD]"
username="[DATABASE_USER_NAME]">
  <server port="[DATABASE_PORT]" url="[DATABASE_SERVER_HOST]"/>
</db>
```

Пример:

```
<dbserver>
  <!--default database instance-->
  <!--connectionValidTime in minutes: time a connection is valid, value 0 means unlimited -->
  <!--connectionIdleTime in minutes: time a connection is valid if not in use, value 0 means unlimited -->
  <db connectionIdleTime="0" connectionValidTime="0" id="default" instance="testinstanz" maxPoolSize="4" minPoolSize="2" password="command" username="command">
    <server port="1521" url="sora01"/>
  </db>
```

Рисунок 4-10 Пример для базы данных

- После успешного обновления базы данных снова запустите экземпляр Tomcat (смотрите раздел **Ошибка! Источник ссылки не найден.**).
- Если обновление прошло без ошибок, Вы можете удалить резервную копию **command_<ГГГГ-ММ-ДД>**.

5 Список изображений

Рисунок 1-1 glm.exe	11
Рисунок 1-2 Веб-интерфейс сервера лицензий	13
Рисунок 1-3 Статус glm	14
Рисунок 1-4 Управление службами Windows для сервера лицензий RLM	15
Рисунок 1-5 Опции для перезапуска	15
Рисунок4-1 Папка для резервного копирования	45
Рисунок 4-2 Пример для порта	46
Рисунок 4-3 Пример для порта SSL	46
Рисунок 4-4 Пример для wmlcons	46
Рисунок 4-5 Пример для базы данных.....	46
Рисунок4-6 Директория для резервного копирования	47
Рисунок4-7 Пример для порта	48
Рисунок 4-8 Пример для порта SSL	48
Рисунок 4-9 Пример для wmlcons	48
Рисунок 4-10 Пример для базы данных.....	48

6 Выходные данные



СДИ софт
Россия, 107045, г. Москва
Трубная ул., 12
Телефон: +7 499 495 1024
Сайт: www.sdisoft.ru
Электронная почта: info@sdisoft.ru

Тексты и иллюстрации были подготовлены с большой тщательностью.

Несмотря на это некоторые сведения могут содержать ошибки или неточности.

Издатель и авторы не несут юридической или какой-либо другой ответственности за ошибочные данные и ущерб, причиненный в результате использования этих данных.

Издатель и авторы рады любым предложениям по улучшению содержания документа и указания на неточности и ошибки.

Товарный знак и копирайт:

© ООО «СДИ софт». Все права сохранены.

Ни одна из частей данного документа не может быть размножена, воспроизведена или переведена на другой язык, включая фотомеханические средства или хранение в электронном виде, без предварительного письменного разрешения компании «СДИ софт». Изменения не исключены.

Microsoft® и Windows® являются зарегистрированными торговыми марками компании Microsoft Corporation.

Все названия программного и аппаратного обеспечения, используемые в данном руководстве, являются зарегистрированными торговыми марками соответствующих производителей.

Произведено: СДИ софт, Москва

Производитель: СДИ софт, Москва