

DATAMONITOR

# **РУКОВОДСТВО СИСТЕМНОГО ПРОГРАММИСТА**

**Листов 15**

2018

АО «Авикомп Сервисез»

## **АННОТАЦИЯ**

Настоящий документ описывает назначение, необходимые для обеспечения работоспособности технические и программные средства, структуру и настройку, проверки работоспособности, а также сообщения, выдаваемые в ходе проверки и работы системы анализа данных.

## Содержание

1	Общие сведения о программе.....	4
1.1	Назначение программы .....	4
1.1.1	Подключение внешних источников данных.....	4
1.1.2	Создание и редактирование OLAP кубов .....	4
1.1.3	Создание и редактирование визуальных представлений данных.....	5
1.2	Требования к техническим средствам .....	5
1.3	Требования к программным средствам.....	5
2	Установка Системы .....	6
3	Проверка программы.....	9
3.1	Проверка .....	9
4	Сообщения системному программисту .....	11
4.1	Сообщения в ходе проверки модуля загрузки данных в систему визуализации .....	11
5	Мониторинг работоспособности компонент Системы .....	12
5.1	Предварительная настройка почтовой службы на сервере .....	12
5.2	Создание скрипта мониторинга.....	13

# 1 Общие сведения о программе

## 1.1 Назначение программы

Основной целью создания Системы является повышение эффективности обработки и анализа информации, содержащейся в различных информационных ресурсах пользователя.

Назначением Системы является формирование интегрированных аналитических данных и их визуализации в интересах принятия управленческих решений должностными лицами Заказчика.

Основными возможностями Системы являются:

Система обеспечивает выполнение следующих функций (задач):

- подключение внешних источников данных (базы данных и структурированные текстовые файлы);
- создание и редактирование OLAP кубов;
- создание и редактирование визуальных представлений данных.

### 1.1.1 Подключение внешних источников данных

Система предоставляет возможность подключения реляционных баз данных, а также настройки и последующей загрузки данных из структурированных текстовых файлов

### 1.1.2 Создание и редактирование OLAP кубов

В рамках задачи создания и редактирования OLAP кубов, Система обеспечивает:

1. Формирование OLAP куба по множеству таблиц и отдельных колонок.
2. Объединение таблиц по внешним ключам (foreign key)
3. Формирование наборов срезов по данным, манипуляцию и трансформацию данных включая:
  - выполнение функций агрегации данных;
  - конкатенации и манипуляции строковыми значениями

- выполнение различных математических операторов над данными
- трансформация типов данных (число-строка, строка-дата и т.д.)
- выполнение условных и логических выражений – if, and, or, not.

### **1.1.3 Создание и редактирование визуальных представлений данных**

Визуальные представления данных позволяют сформировать на экране компьютера интегрированные представления множества информационных блоков.

Система позволяет встраивать отдельные информационные блоки в сторонние сайты и web приложения.

## **1.2 Требования к техническим средствам**

Минимально возможными характеристиками системы является:

- процессор Intel Core i3;
- оперативная память - 4 Гб;
- накопители на жестких магнитных дисках объемом 300 Гб.

## **1.3 Требования к программным средствам**

Для обеспечения функционирования панели администратора должны использоваться следующие программные средства:

- операционная система Ubuntu, версии не ниже 16.04;
- СУБД PostgreSQL, версия не ниже 9.4;
- Java JRE 1.8.

Для обеспечения взаимодействия с Системой должен быть установлен браузер:

- Mozilla Firefox;
- Google Chrome.

## 2 Установка Системы

Все действия по развёртыванию Системы выполняются от имени суперпользователя root.

В домашнем каталоге пользователя операционной системы, от имени которого будет осуществляться запуск приложения, создать папку ps\_avi и перейти в неё:

```
mkdir -p /home/<имя_пользователя>/ps_avi  
cd /home/<имя_пользователя>/ps_avi/
```

Скопировать и распаковать дистрибутивный архив ps-avi.tar.gz в текущую директорию:

```
tar -xzf ps-avi.tar.gz
```

Сделать исполняемым стартовый скрипт start.sh и start\_release.sh:

```
chmod u+x jsb-application/start.sh  
chmod u+x jsb-application/start_release.sh
```

Поместить стартовый скрипт keshif-demo в /etc/init.d, установить права и включить в автозапуск:

```
cp keshif-demo /etc/init.d  
chmod a+x /etc/init.d/keshif-demo  
chown root:root /etc/init.d/keshif-demo  
chkconfig keshif-demo on
```

Настроить скрипт /etc/init.d/keshif-demo, исправив в нём значение переменной USER на имя пользователя, в домашней директории которого установлена программа модуля.

Конфигурационные файлы, относящиеся к серверной части программы находятся в директории jsb-application/config.

При необходимости можно выполнить следующие настройки:

- указать номер порта, который слушает серверная часть программы, в файле web.conf, где содержимое файла web.conf:

```
web = {
```

```

folder = "../web"
http = {
port = 8888

```

```

# increase requestHeaderSize in case of long cross-domain GET requests
requestHeaderSize = 131072

```

```

# increase responseBufferSize in case of large response

```

```

# responseBufferSize = 67108864
}

```

```

widgets = {
path = "widgets"
}

```

```

uploadCacheFolder = ".fileUploadCache"

```

```

debug = false

```

```

}

```

- для изменения номера порта, который слушает серверная часть программы, необходимо заменить значение в поле `http-port = 8888` этого файла;

- установить владельцем директории `ps_avi` пользователем, в домашней директории которого установлен модуль:

```

chown -R <имя_пользователя>:<группа_пользователя>
/home/<имя_пользователя>/ps_avi

```

Например:

```

chown -R user:user /home/user/ps_avi

```

Запуск:

```

service keshif-demo start

```

```

service keshif-demo status

```

[ ok ] keshif-demo is running.

Проверить, что Система стартовала, можно перейдя по ссылке:  
<http://<Имя хоста на котором установлен модуль>:8888/datamonitor.jsb>



### 3 Проверка программы

Для проверки корректности установки программы осуществите вызов Системы, выполнив следующие действия:

Запустите установленный на рабочей станции пользователя веб-браузер.

Наберите адрес Системы в строке вызова и нажмите [Enter].

#### 3.1 Проверка

Из домашней директории пользователя ОС Ubuntu, под которым устанавливается программа модуля, перейти в папку `./ps_avi_apps/keshif-demo/jsb-application`, в которой находятся исполнимые файлы и папки проекта модуля:

**`cd ./ps_avi_apps/keshif-demo/jsb-application/`**

Выполнить командный скрипт `start.sh`. Командный скрипт `start.sh` инициирует выполнение программы модуля. Вывести на экран журнал выполнения программы модуля из файла `nohup.out` командой

**`less +F nohup.out`**

Найти в результатах вывода журнала строки, аналогичные следующим строкам:

- **[INFO] [08/07/2017 10:38:13.946] [jsBeans-akka.actor.default-dispatcher-10] [akka://jsBeans/user/JsHub] Service JsHub initialized - 0.073 s**
- **2017-08-07 10:38:21,530 [INFO] (org.jsbeans.scripting.JsBridge.JS:(root):257) - Images has been successfully loaded**

Наличие этих строк свидетельствует, что серверная часть программы стартовала корректно.

После старта серверной части программы пользователь должен стартовать веб-браузер FireFox с адресной строкой, содержащей

обращение к серверу программы. Адресная строка, содержащая обращение к серверу программы, имеет вид:

**http://<ip>:<номер порта>/datacube/Editor.jsb**

Тут <ip> и <номер порта> – соответственно ip-адрес сервера, на котором установлен модуль, и номер порта, указанный в конфигурационном файле web.conf, например:

**http://localhost:8888/datacube/Editor.jsb**

Если пользователь закрывает окно веб-браузера с компонентами визуального интерфейса программы, то серверная часть программы не останавливается и продолжает работу.

## 4 Сообщения системному программисту

Система выводит сообщения о ходе работы в протокольные файлы.

### 4.1 Сообщения в ходе проверки модуля загрузки данных в систему визуализации

Сообщения, формируемые Системой в ходе проверки модуля загрузки данных в систему визуализации, представлены в таблице 1.

Табл. 1 – Сообщения модуля загрузки данных в систему визуализации

№ п/п	Пример сообщения	Описание сообщения	Компонент Модуля, где выдается сообщение
1.	Error while reading data stream	Ошибка чтения данных от интерфейсной части в случае передачи потоковой передачи тела HTTP запроса	Сервер
2.	Request data is invalid	Ошибка получения данных запроса сервером	Интерфейс
3.	Data .. is not a valid JSON	Ошибка разбора данных сервером, когда полученные данные не являются данными в формате JSON	Интерфейс
4.	<тип параметра>... is not setted for ... <тип команды>	Ошибка разбора данных сервером, когда в списке параметров отсутствует необходимый тип параметра	Интерфейс
5.	Sending POST request to ... Response code : ... Authorizing to ... with login ... Checking cube datasource	Сообщения, формируемые при работе с сервером системы визуализации	Сервер
6.	Getting indicator types tree... Indicator template saving Connecting to db... Getting indicator type data	Сообщения, формируемые на старте выполнения запросов	Сервер
7.	Требуется ... параметр	Всплывающие сообщения при отсутствии данных в необходимых полях формы ввода данных	Интерфейс

## 5 Мониторинг работоспособности компонент Системы

### 5.1 Предварительная настройка почтовой службы на сервере

Для предварительной настройки почтовой службы на сервере необходимо выполнить следующие действия:

1. Установить пакеты:

```
apt-get install mutt exim4-base exim4-config exim4-daemon-light
```

2. Выполнить настройку exim4, пример:

```
dpkg-reconfigure exim4-config
```

Общий тип почтовой конфигурации: *отправка почты через smarthost; приём по SMTP или через fetchmail*

Почтовое имя системы: *указать имя сервера, например srv1.example.net*

IP-адреса, с которых следует ожидать входящие соединения SMTP: *127.0.0.1*

Другие места назначения, для которых должна приниматься почта: *указать имя сервера, например srv1.example.net*

Машины, для которых доступна релейная передача почты: *Пусто*

IP-адрес или имя хоста являющегося исходящим smarthost: *указать адрес smtp шлюза*

Скрывать локальное почтовое имя в исходящей почте? *Нет*

Сокращать количество DNS-запросов до минимума (дозвон по требованию)? *Нет*

Метод доставки локальной почты: *mbox формат в /var/mail/*

Разделить конфигурацию на маленькие файлы? *Нет*

3. Будет выполнен автоматический рестарт MTA:

*[ ok ] Stopping MTA for restart:.*

*[ ok ] Restarting MTA: exim4.*

## 5.2 Создание скрипта мониторинга

1. Создать скрипт, выполняющий стандартную проверку статуса сервиса, ps\_check.sh:

```
#!/bin/bash

ret=0
message=""
services="keshif-demo ontoed-d2rq ontoed-engine passport-web-service
unstructured-data-api unstructured-data-uploader ogcserver"
for i in $services; do
    state=$(/usr/sbin/service $i status)
    reti=$?
    message="$message\n$state"
    let ret=$ret+$reti
done
test $ret -ne 0 && { echo -e "$message" | /usr/bin/mutt -s "Уведомление о
проверке статуса приложений" -- admin@eip.ru admin2@eip.ru; }
```

2. Поместить скрипт в каталог /usr/local/bin/ и установить флаг исполняемый и владельца:

```
chmod 750 /usr/local/bin/ps_check.sh
chown root:root /usr/local/bin/ps_check.sh
```

3. Настроить автозапуск по расписанию "crontab -e root", добавить строку:

```
*/5 * * * * /usr/local/bin/ps_check.sh >/var/log/ps_check.log 2>&1
```

4. В процессе проверки статуса сервиса, в случае ошибки (код возврата не равен 0), будет инициирована отправка сообщения на указанные адреса.

Для правильного функционирования проверки, необходимо использовать поставляемые в комплекте с ПО стартовые init.d скрипты.

Результатом проверки, в случае обнаружения ошибки, будет примерно такое письмо админу:

***От: root <root@test-ald-client.private>***

***Кому: techno@avicomput.ru***

***Тема: Уведомление о проверке статуса приложений***

***Дата: Mon, 18 Sep 2017 16:35:01 +0300***

***keshif-demo is not running ... failed!***

***ontoed-d2rq is not running ... failed!***

***ontoed-engine is running.***

***passport-web-service is running.***

***unstructured-data-api is not running ... failed!***

***unstructured-data-uploader is running.***

5. Аналогично в список проверяемых компонент добавить postgresql, apache2 и т.д.

## ПЕРЕЧЕНЬ ПРИНЯТЫХ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

Сокращение	Расшифровка
JSON	JavaScript Object Notation -текстовый формат обмена данными, основанный на JavaScript
URL	Uniform Resource Locator - единый указатель ресурса
XML	eXtensible Markup Language— расширяемый язык разметки
БД	База данных
ЛВС	Локальная вычислительная сеть
ОЗУ	Оперативное запоминающее устройство
ОПО	Общее программное обеспечение
ОС	Операционная система
ПЗУ	Постоянное запоминающее устройство
ПО	Программное обеспечение
СУБД	Система управления базами данных