Backend

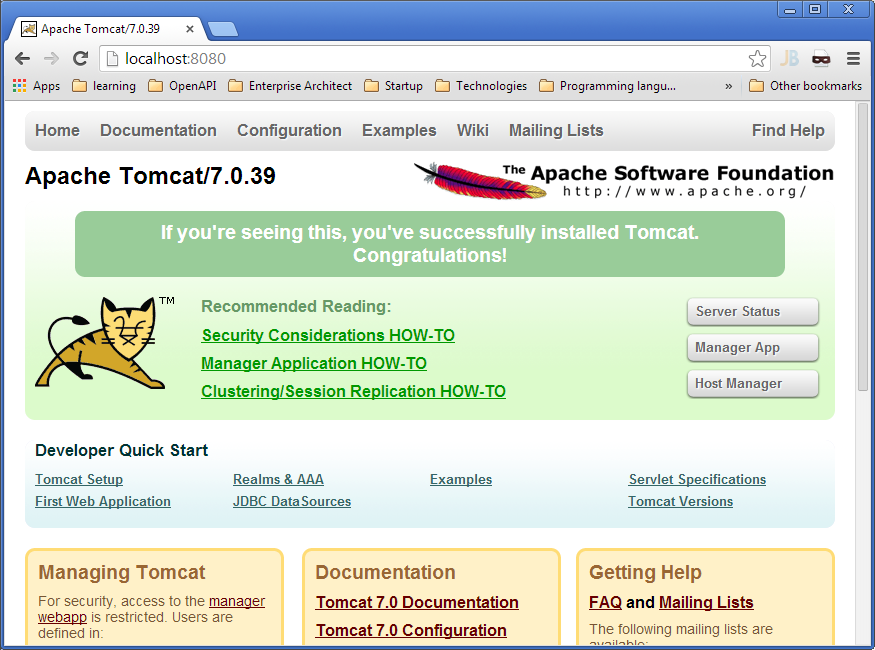
Инфраструктура Open API выдвигает ряд требований к бэкенду (**prod 48350**):

* Трассировка
* Формат данных
* Регистрация модуля в Zookeeper
* Авторизация
* Работа с предикатами

В качестве примера для установки и настройки, мы будем использовать подсистему OAPI\_BIS\_API\_BACKEND. Подсистема предоставляет доступ к функциям продукта BIS и реализована в виде Java веб приложения, работающего на сервере Tomcat.

Установка OAPI\_BIS\_API\_BACKEND

1. Установить Tomcat сервер:
   1. Распаковать файл **apache-tomcat-7.0.39.tar** в любую папку (путь не должен содержать пробелов и русских букв)
   2. Запустить файл **bin\startup.bat**, в случае возникновения ошибок, проверить задана ли переменная JAVA\_HOME (JAVA\_HOME=C:\Program Files\Java\jdk1.7.0\_45)
   3. Открыть страницу <http://localhost:8080>



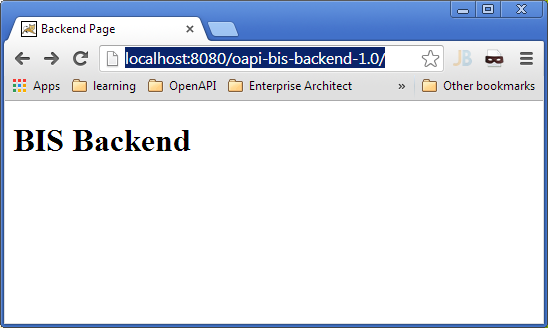
1. Распаковать файл oapi\_bis\_api\_backend-usr-xxx.xx.zip и скопировать файл **oapi-bis-backend-1.0.war** в папку **webapps/** в Tomcat. **(При старте, приложение зачитает параметры подключения к БД из ZooKeeper. Параметры должны быть прописаны!!!)**
2. Запустить Tomcat сервер. При старте произойдет deploy приложения и в папке webapps создастся папка oapi-bis-backend-1.0. После этого остановить сервер.
3. Выполнить рад настроек приложения:
   1. В файле webapps\oapi-bis-backend-1.0\WEB-INF\classes\**app.properties** задать параметр **secretKey=1111** (Ключ шифрования пароля к бд, который храниться в zookeeper)
   2. В файле webapps\oapi-bis-backend-1.0\WEB-INF\classes\**applicationContext.xml** изменить значение bean isIdmWork на **false** (было true)

|  |
| --- |
| <bean id="isIdmWork" class="java.lang.String">  <constructor-arg><value>**false**</value></constructor-arg>  </bean> |

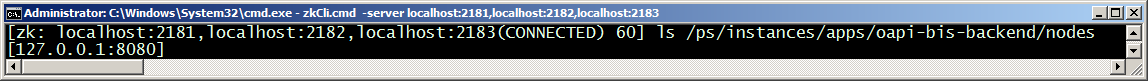
* 1. В файле webapps\oapi-bis-backend-1.0\WEB-INF\classes\**dbContext** изменить количество коннекций к бд - значение maxActive на **5** (было 60)

|  |
| --- |
| <bean id="oraDataSource" class="org.apache.tomcat.jdbc.pool.DataSource" lazy-init="false" destroy-method="close">  ...  <property name="maxActive" value="**5**"/>  <property name="initialSize" value="1"/>  <property name="validationQuery" value="select 1 from dual"/>  <property name="validationInterval" value="10000"/>  <property name="testOnBorrow" value="true"/>  </bean> |

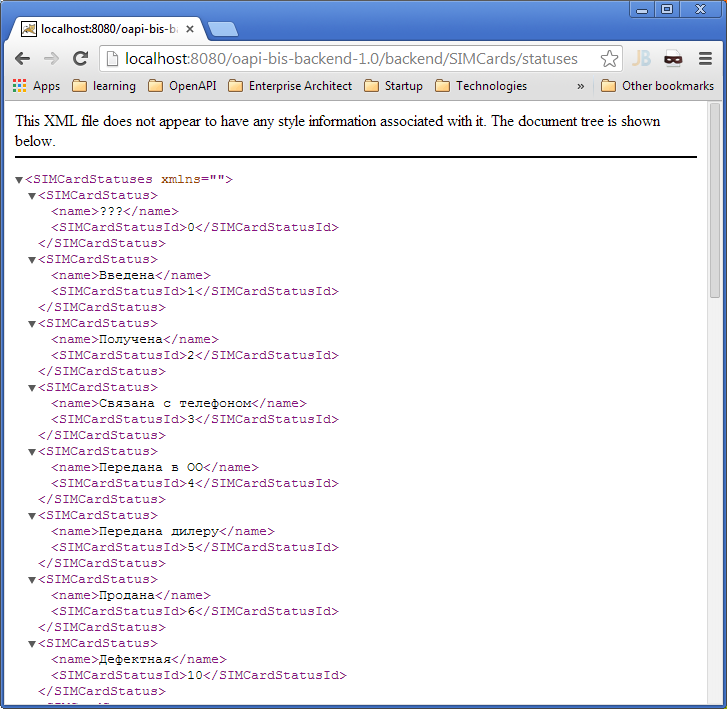
1. Перезапустить Tomcat. Открыть страницу <http://localhost:8080/oapi-bis-backend-1.0>



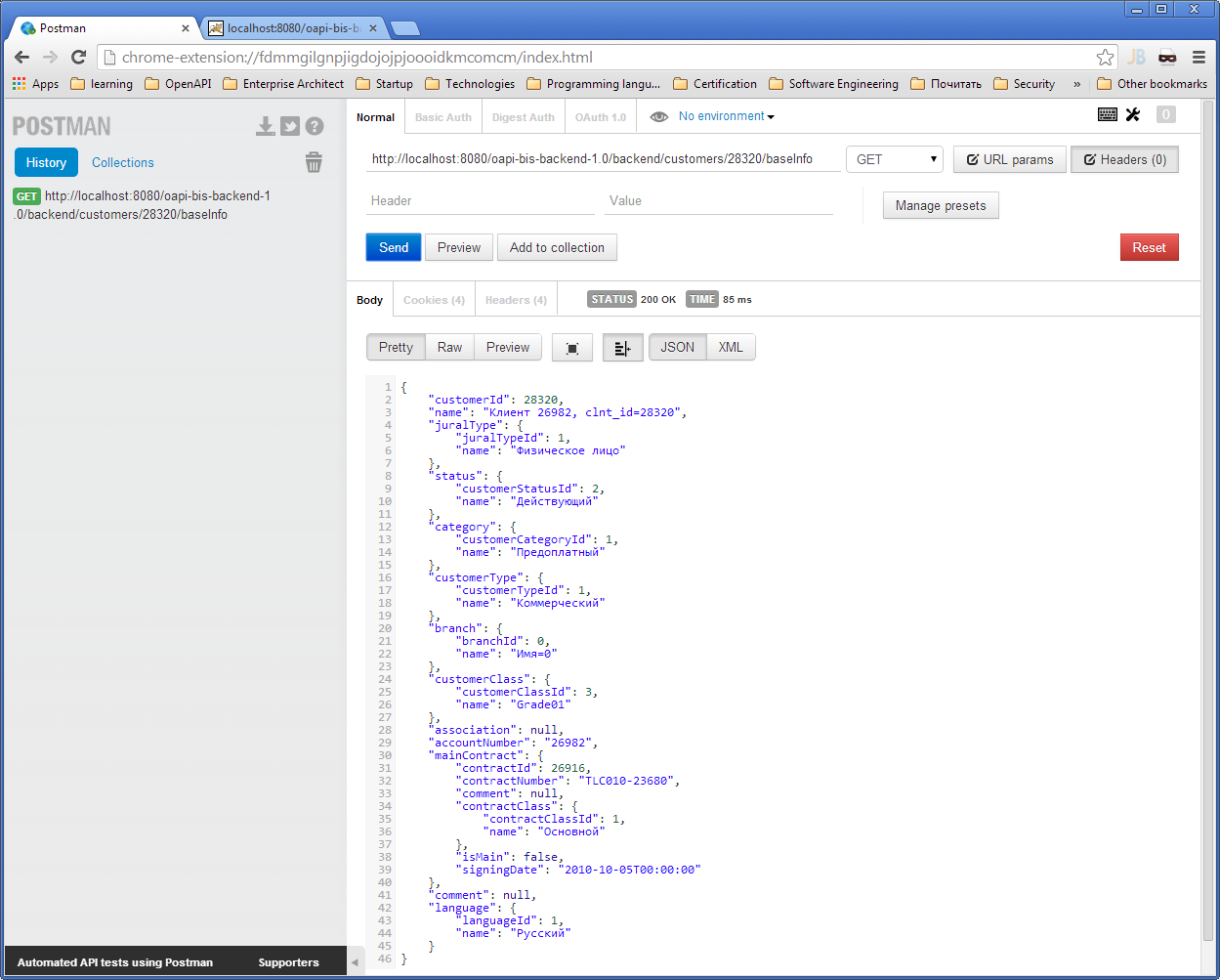
1. Проверить, что приложение зарегистрировало себя в ZooKeeper:



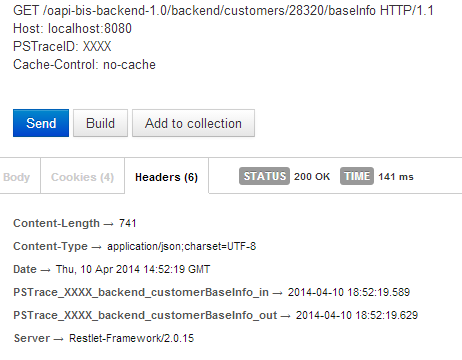
1. Выполнить запрос <http://localhost:8080/oapi-bis-backend-1.0/backend/SIMCards/statuses>



1. Для тестирования RESTful API удобно использовать REST клиент, который позволяет указывать метод HTTP запроса, задавать заголовки и сохранять историю запросов. В качестве примера, можно использовать плагин для браузера chrome: <https://chrome.google.com/webstore/detail/postman-rest-client/fdmmgilgnpjigdojojpjoooidkmcomcm> или любой другой клиент.



1. С помощью следующих запросов определить имя клиента, владеющего номером телефона 89211000300
   1. [http://localhost:8080/oapi-bis-backend-1.0/backend/customers/byMSISDN/{msisdn](http://localhost:8080/oapi-bis-backend-1.0/backend/customers/byMSISDN/%7Bmsisdn)}
   2. http://localhost:8080/oapi-bis-backend-1.0/backend/customers/{customerId}/baseInfo
2. Согласно требованиям к бэкенду, по трассировке, при передаче в запросе заголовка PSTraceID=XXXX, в ответе должны присутствовать заголовки: PSTrace\_XXXX\_backend\_customerBaseInfo\_in и PSTrace\_XXXX\_backend\_customerBaseInfo\_out

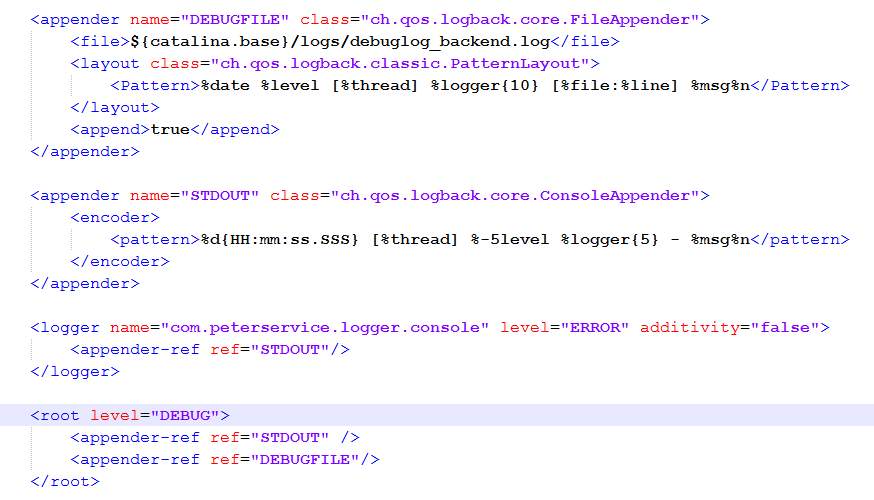


Логирование

Логирование подсистемы OAPI\_BIS\_API\_BACKEND настраивается в файле

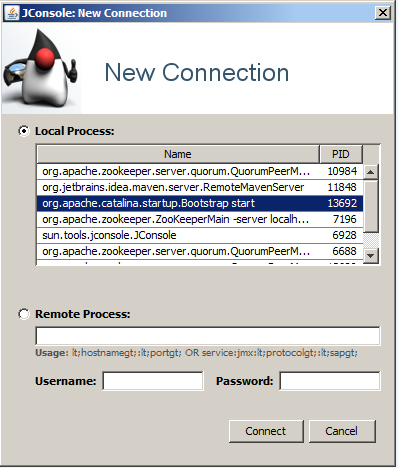
webapps\oapi-bis-backend-1.0\WEB-INF\classes\**logback.xml**

Например, для повышения уровня логирования до уровня DEBUG и включения записи в лог файл можно изменить тег <root>, следующим образом:

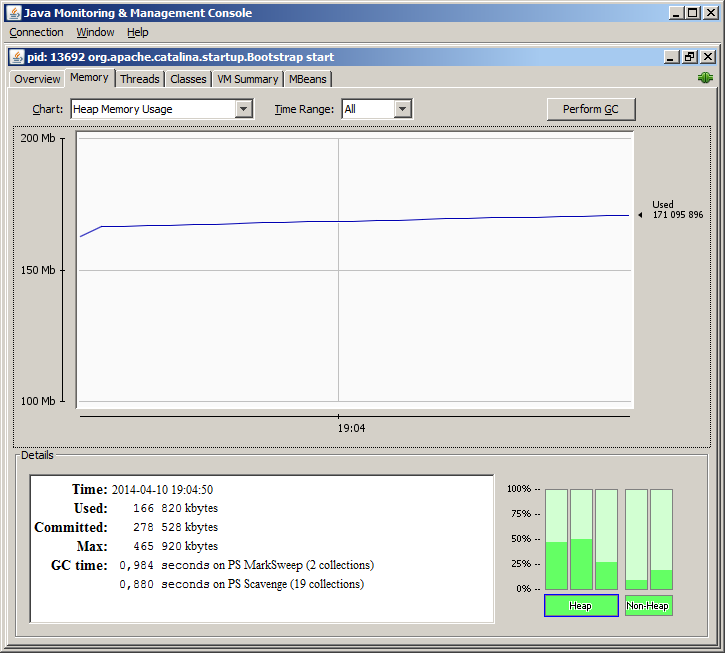


Мониторинг

1. Открыть **jconsole** (%JAVA\_HOME%\bin\jconsole.exe)
2. Выбрать локальный процесс Tomcat:



Потребление памяти Tomcat сервер:



Доступные маршруты, реализованнные с помощью фреймворка Camel (т.к. сервисы реализованы с с помощью Camel, это список доступных сервисов приложения):

