

Корпоративный информационный портал, который построил Джек

Антон Борисов

Представьте себе компанию, перешедшую с бумажного на электронный документооборот. Дайте ей фору в несколько лет, чтобы она обзавелась электронной почтой, разместила сайт в Сети, в общем и целом, увязла в электронной паутине. Дайте ей фору еще в несколько лет, и окажется, что активно идут попытки консолидировать ту информацию, что хранится внутри компании, и получаемую извне. Для вас это уже актуально? Тогда – полный вперед!

Исторические предпосылки

В середине 90-х годов прошлого века мы запомнили такие веб-порталы, как Excite, Webcrawler, Yahoo!, Lycos, Altavista. Эти сайты предоставляли такую информацию, как новости, погоду, биржевые сводки, поиск, а также ряд других сервисов. Зачастую она была представлена в виде отдельных веб-приложений (портлетов). По мере того как предприятия переходили на электронный документооборот, росла и их потребность в таких же порталах, что имеют Yahoo! и им подобные, но в рамках данного конкретного предприятия.

Корпоративный информационный портал – это место для интеграции информации, процессов и сообщества людей в рамках организации. Отличительными чертами порталов являются: децентрализованные источники информации, система управления содержанием (Content Management System). За счет того, что портал предоставляет унифицированную точку доступа к информации, появляется возможность настроить получение информации в той форме, которая наиболее удобна пользователю.

Портал может состоять из следующих модульных подсистем:

- **визуальный слой** – интерфейс веб-пользователя;
- **персонализация** – способность формировать динамический ответ пользователю в зависимости от его персонального профиля;
- **средства совместной работы** – инструменты для обмена информацией между пользователями (электронная почта, общие ресурсы и т. д.);
- **портлеты** – структуры, предназначенные для легкого подключения программных модулей и сервисов приложений;
- **поиск и навигация** – когда содержание портала скомпоновано по категориям и предоставлено средство поиска необходимой информации;
- **публикация и подписки** – возможность авторизовать новое содержание и публиковать его подписчикам;
- **администрирование и безопасность** – планировщик страниц, средства мониторинга метаданных.

В конце 90-х многие компании-разработчики ПО начали предлагать решения в виде порталов для корпоративных услуг. Ряд из них был разработан на определенной программной платформе, поэтому, выбирая то или иное решение, компания не могла взять бизнес-логику от другого портала – приходилось с нуля проектировать нужное приложение.

Программная платформа могла базироваться как на открытых технологиях, так и на закрытых, с коммерческим лицензированием или лицензи-



Рисунок 1. Структурная схема портала, созданного из портлетов

рованием в соответствии с GPL/LGPL/MIT/Apache License. Примерами могут служить:

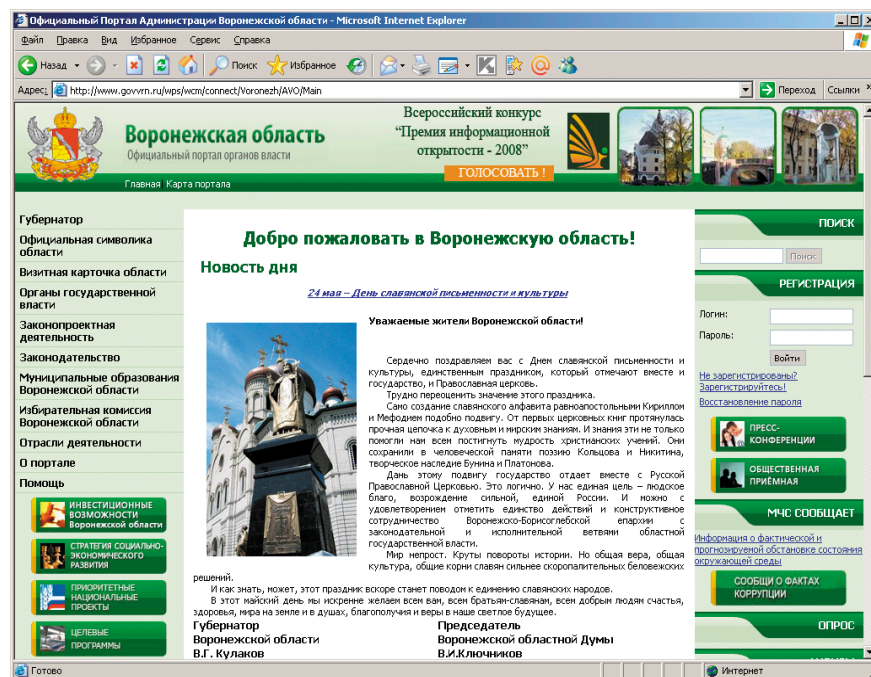


Рисунок 2. Портал администрации Воронежской области

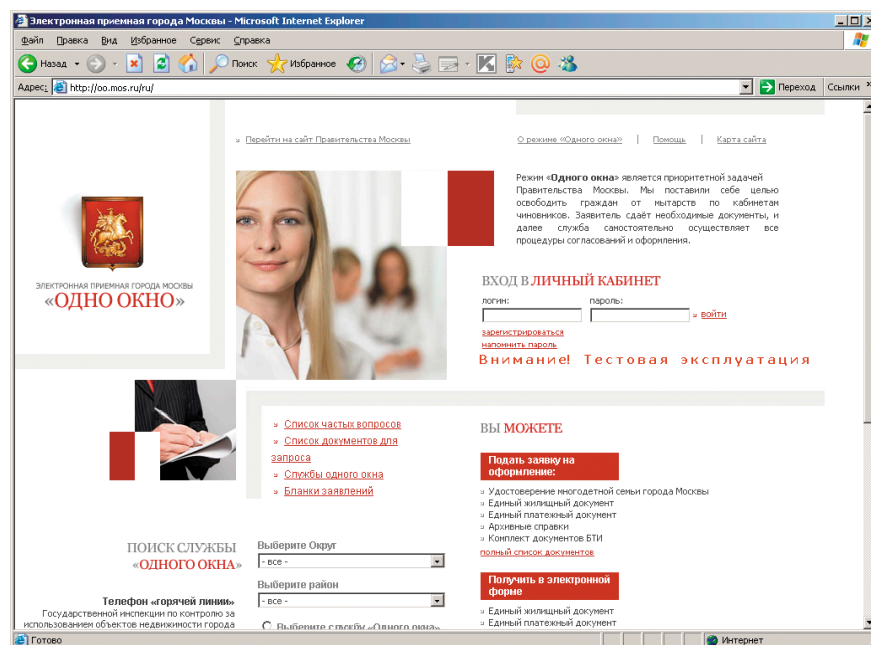


Рисунок 3. Исполнительная власть Москвы реализует программу «Одно Окно»

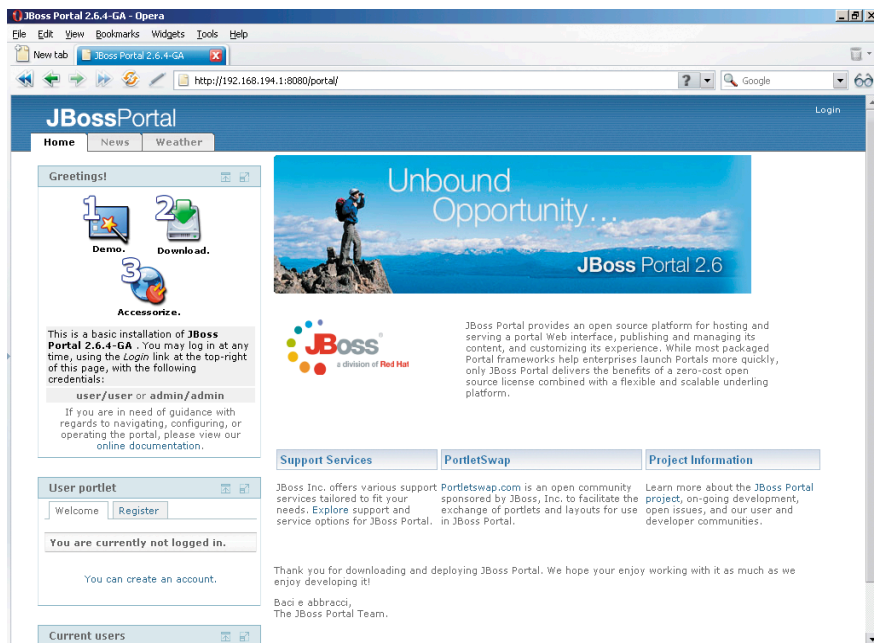


Рисунок 4. JBoss Portal успешно развернут

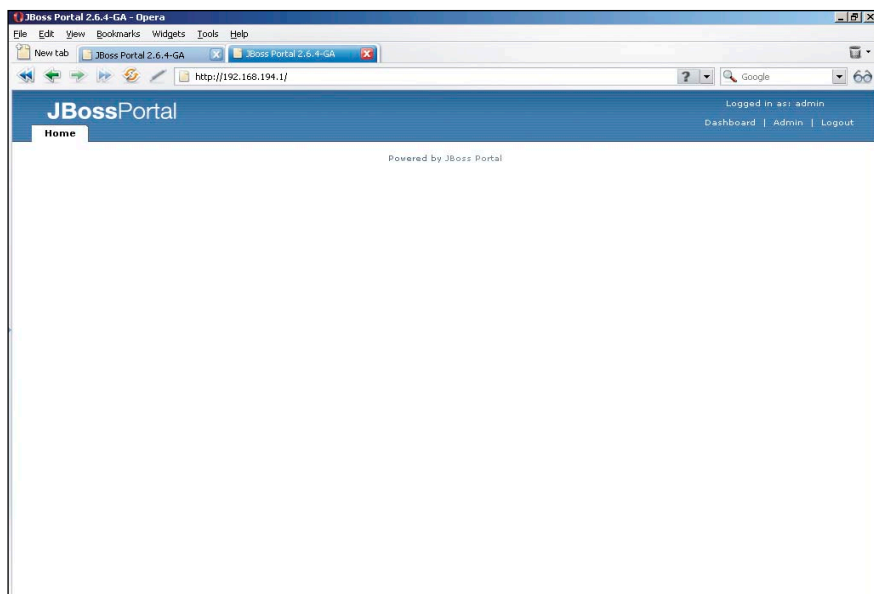


Рисунок 5. Может показаться, что за внешней простотой скрывается простой механизм, однако это не так

- WebSphere Portal Server – J2EE/коммерческое лицензирование;
- Microsoft Office Sharepoint Server – ASP.NET/коммерческое лицензирование;
- Oracle Application Server Portal – J2EE/коммерческое лицензирование;
- Sun Java System Portal Server – J2EE/смешанное лицензирование;
- Jetspeed – J2EE/Apache License;
- JBoss Portal – J2EE/LGPL;
- Liferay Portal – J2EE/MIT.

Однако в 2003 году появляется инициатива использовать единый стан-

дарт (Java Specification Request 168 или JSR168) для корпоративных порталов, построенных на основе Java. Цель данного стандарта – предоставить единый API для взаимодействия между порталами и портлетами. Поэтому если вы запустили на предприятии портал, который поддерживает стандарт JSR168, то у вас высокие шансы, что портлеты, созданные в соответствии с данным стандартом, заработают и у вас.

Что же такое портлет на самом деле? В первую очередь это веб-приложение, которое запущено на сервере приложений Java, например Tomcat. И во-вторых, это компонент, который

встраивается в структуру пользовательского интерфейса портала. Выберет ли пользователь, что на его персонализированной странице портала должны высвечиваться биржевые сводки или нет, зависит полностью от его решения (см. рис. 1).

Сферы применения

Назначение порталов может быть различным. Это может быть портал компании, портал органа власти, портал госуслуг, портал вуза. Все они призваны служить какой-либо определенной цели, которые детально описаны по ссылке [1]. Приведу лишь примеры реализации порталов.

Корпоративные порталы:

- ОАО «Вымпелком» – <http://www.msk.beeline.ru>;
- Steelcase Inc. – <http://www.steelcase.com>;
- Siemens AG – <http://siemens.com>.

Порталы органов власти:

- Еврейская автономная область – <http://www.eao.ru>;
- администрация Воронежской области – <http://www.govvrn.ru>;
- администрация Тульской области – <http://admportal.tula.ru>;
- Министерство экономического развития РФ – <http://www.economy.gov.ru>.

Порталы госуслуг:

- Федеральное агентство по техническому регулированию и контролю – <http://www.gost.ru>;
- электронная приемная города Москвы («Одно Окно») – <http://oo.mos.ru>;
- проект портала Зеленограда с его мобильными услугами – <http://zelao.mobi>;
- портал госуслуг Республики Киргизия – <http://www.govservices.kg>.

Порталы вузов:

- Брянская сельскохозяйственная академия – <http://www.bgsha.com>;
- Санкт-Петербургский университет информационных технологий, механики и оптики – <http://www.ifmo.ru>;
- портал информационной поддержки ЕГЭ – <http://ege.edu.ru>;
- портал непрерывного образования преподавателей – <http://www.neo.edu.ru>;

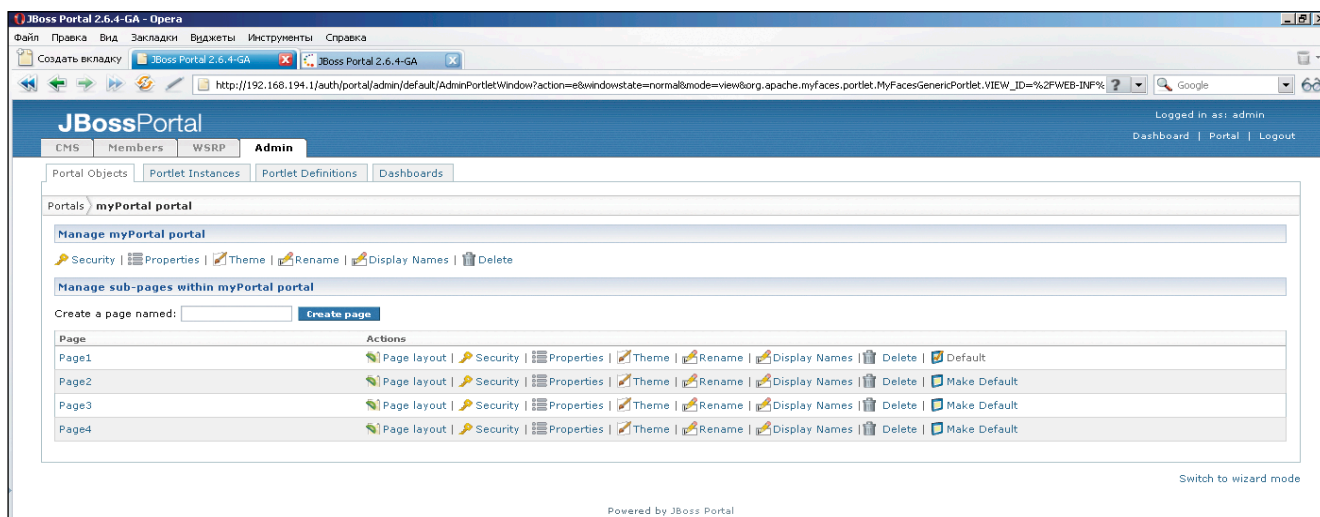


Рисунок 6. В административном режиме начинаем формировать свой портал

- Казанский государственный технический университет – <http://www.kai.ru>;
- Академия управления ТИСБИ – <http://www.tisbi.ru>.

В качестве международного портала стоит взглянуть на пример Индии – <http://india.gov.in> и на пример поискового портала – <http://lycos.com>.

Что интересно, у такой разноплановой компании, как Microsoft, есть рекомендательный документ, как следует строить порталы, в частности, портал органов государственной власти [2].

Проблема выбора

Почему был выбран именно JBoss Portal? На то было несколько причин:

- хотелось отказаться от привязки к какой-либо конкретной операционной системе;
- иметь поддержку Java-приложений;
- иметь программное решение на открытых стандартах;
- иметь возможность хранить данные в различных СУБД.

Для первых трех пунктов в качестве аргумента выступила связка JAVA + JSR168. Что касается СУБД, то работа со следующими базами – MySQL, PostgreSQL, Oracle, MSSQL, HyperSonic DB – также оказалась достаточной.

Ну и напоследок: за спиной JBoss стоит компания RedHat, которую вряд ли можно соблазнить эфемерными проектами. В 2006 году компания JBoss была куплена последней и преобразована в структурное подразделение – RedHat Middleware LLC. Напомню, что на протяжении уже больше десяти лет RedHat работает флагманом, который медленно, но верно двигает информационный рынок в части Linux-решений. Поэтому есть оп-

ределенные предпосылки, что порталное решение будет развиваться и в будущем. Этот факт тоже внес свою лепту в выбор JBoss Portal.

Установка портала

Итак, забираем дистрибутив с JBoss Portal [3], создаем в системе пользователя с именем jboss. Распаковываем, например, в директорию jboss-portal. Впоследствии при упоминании JBOSS_HOME будем знать, что подразумевается директория /home/jboss/jboss-portal. Но перед стартом проведем несколько операций с СУБД. В данном примере выбран MySQL.

Шаг 1. Создаем в MySQL-хранилище новую базу с именем html_data:

```
mysql> create database html_data charset = utf8;
```

Шаг 2. Для пользователя jbossuser разрешаем доступ к этой базе, как с локального хоста, где крутится JBoss

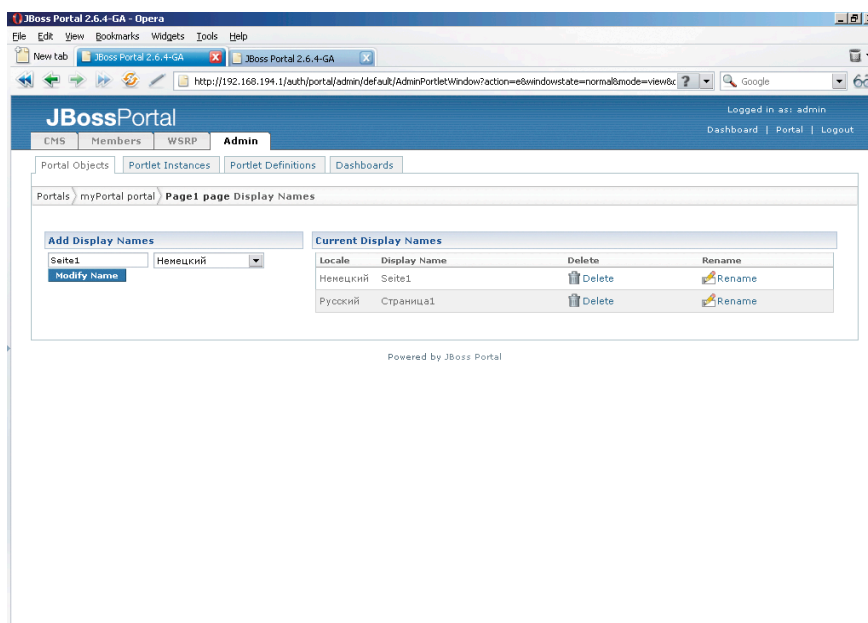


Рисунок 7. Для каждой языковой группы посетителей можно назначить нужные названия страниц

Portal, так и с рабочего места, откуда будут вноситься изменения в структуру базы:

```
mysql> grant select, insert, update, delete, create,  
drop on html_data.* to 'jbossuser'@'localhost';
mysql> grant select, insert, update, delete, create,  
drop on html_data.* to 'jbossuser'@'192.168.194.128';
mysql> grant select, insert, update, delete, create,  
drop on html_data.* to 'jbossuser'@'192.168.194.1'  
identified by 'jboss';
```

Шаг 3. Зайдем в MySQL-хранилище под именем пользователя jbossuser:

```
ab@abc:~> mysql -u jbossuser html_data
```

Шаг 4. И создадим таблицу с названием news:

```
mysql> create table news  
(id integer not null auto_increment, `text` text,  
primary key (id)) charset = utf8;
```

Шаг 5. Затем с рабочего места заполним эту таблицу значениями с помощью SQLyog.

Шаг 6. Далее следует в JBoss Portal выставить, что следует использовать MySQL-хранилище, т.к. по умолчанию будет использоваться база в формате Hypersonic DB.

Для этого из директории \$JBOSS_HOME/setup/ следует скопировать файл portal-mysql5-ds.xml в директорию \$JBOSS_HOME/server/default/deploy/. Файл, который используется по умолчанию, portal-hsqldb-ds.xml, нужно либо удалить, либо переименовать как portal-hsqldb-ds.xml.original.

Затем в файле portal-mysql5-ds.xml следует поменять имя пользователя и его пароль, от имени которого будет создана порталная база. Примерное содержимое файла:

```
<datasources>
<local-tx-datasource>
<jndi-name>PortalDS</jndi-name>
<connection-url>jdbc:mysql://127.0.0.1:3306/  
jbossportal?useServerPrepStmts=false&  
jdbcCompliantTruncatio
n=false</connection-url>
<driver-class>com.mysql.jdbc.Driver</driver-class>
<user-name>jbossdb</user-name>
<password>jbossdb</password>
</local-tx-datasource>
</datasources>
```

Ключевые параметры выделены красным шрифтом.

Не забудем также взять с сайта MySQL.com jdbc-коннектор [4] и скопировать его в \$JBOSS_HOME/server/default/lib/mysql-connector-java-5.1.6-bin.jar. Обращаю ваше внимание на то, что таблица в MySQL-базе хранится в UTF8-форме (см. «create table ... charset = utf8»). Это связано с тем, что внутри JBP используется несколько языковых интерфейсных схем, в частности испанская и французская. И чтобы минимизировать проблемы с перекодировкой между порталом и браузерами, было решено хранить в UTF8. Еще одно замечание – проследите, чтобы в файле my.cnf было явно указано, какая кодировка используется:

```
[mysqld]
character_set_server=utf8
```

В частности, моя рабочая система работает с такими параметрами:

```
mysql> show variables like 'character%';
```

Variable_name	Value
character_set_client	latin1
character_set_connection	latin1
character_set_database	utf8
character_set_filesystem	binary
character_set_results	latin1
character_set_server	utf8
character_set_system	utf8
character_sets_dir	/usr/share/mysql/charsets/

8 rows in set (0.00 sec)

```
mysql> show variables like 'collation%';
```

Variable_name	Value
collation_connection	latin1_swedish_ci
collation_database	utf8_general_ci
collation_server	utf8_general_ci

3 rows in set (0.00 sec)

И еще один шаг перед запуском портала. Дело в том, что JBoss Portal хранит в базе данных не только имена пользователей портала, их электронные адреса и другую служебную информацию, но и содержимое CMS (HTML-файлы). Ясно, что по мере того как портал будет заполняться новой информацией (читай, новыми HTML-страницами), то будет расти и размер базы. Но есть вариант, использовать внешнее хранение – непосредственно на диске, а не в базе. Для этого изменим значения параметров externalBLOBs в файле \$JBOSS_HOME/server/default/deploy/jboss-portal.sar/portal-cms.sar/META-INF/jboss-service.xml с «false» на «true»:

```
<param name="externalBLOBs" value="true"/>
```

Шаг 7. Запускаем портал:

```
$ $JBOSS_HOME/bin/run.sh --host 192.168.194.1
```

Как видите, для доступа к portalу нужно вводить URI в виде http://192.168.194.1:8080/portal/, что представляется не совсем удобным. Варианты следующие: включить перенаправление входящих запросов с порта 80 на локальный порт 8080 и изменить расположение доступа к portalу – Context Path:

```
# /usr/sbin/iptables -t nat -A PREROUTING -i eth0  
-p tcp --dport 80 -j REDIRECT --to-port 8080
```

И поменяем Context Path в файле \$JBOSS_HOME/server/default/deploy/jboss-portal.sar/portal-server.war/WEB-INF/jboss-web.xml.

Новое значение будет выглядеть как:

```
<context-root>/</context-root>
```

Зайдя на адрес 192.168.194.1, вы увидите стандартное окно JBoss Portal.

Создаем простейшее представление

Первым делом попробуем составить свой собственный портал, немного отличный от стандартного.

Для этого заходим под пользователем admin (его имя и пароль высвечиваются в портлете Welcome – название шапки «Greetings!»). Следуем в меню «Admin», что находится среди Dashboard, Admin, Logout, и во вкладке «Portal Objects» увидим, что существует 3 портала – admin, default и template. Причем портал default выбран текущим, что и понятно из названия и пометки в виде галочки.

Вводим название создаваемого портала, например myPortal. Нажимаем на кнопку «Create Portal» и затем нажимаем на ссылку «Make Default». Зайдя по адресу <http://192.168.194.1>, можно полюбоваться на новый пустой портал (см. **рис. 5**).

Первым делом попробуем создать несколько страниц, которые наполним текстом и добавим на них несколько портлетов. Заодно и поменяем тему, чтобы можно было отличить наш портал от тысячи таких же, построенных на JBoss Portal.

Создадим 4 страницы с названиями: «Страница1», «Страница2», «Страница3» и «Страница4». Лучше писать их названия в латинской раскладке, а затем назначить русский перевод.

Итак, выбираем нужный портал – myPortal. В этом окне в поле «Create a page named:» набираем Page1 и подтверждаем, нажав на кнопку «Create page». Аналогично для остальных страниц.

Первую страницу делаем страницей по умолчанию. Зайдем на вкладку для страницы Page1, названную «Display Names». Здесь для каждого языка можно сделать свой собственный перевод названия страницы. Зачем это нужно? Представьте, что на страницы портала заходят посетители из различных языковых групп, в качестве примера: из Франции, Германии, США и русскоязычные пользователи. Чтобы каждому из них страница показывалась не только как «Page1», но и как «Seite1», «La Page1», «Страница1», и придуман данный раздел.

Необходимо заполнить поле с названием и выбрать язык локализации. Внимание, если вы ввели локализацию для русского языка, а в браузере клиента она не будет использоваться, то для сложных порталовых решений возможны случаи, когда портал пока-

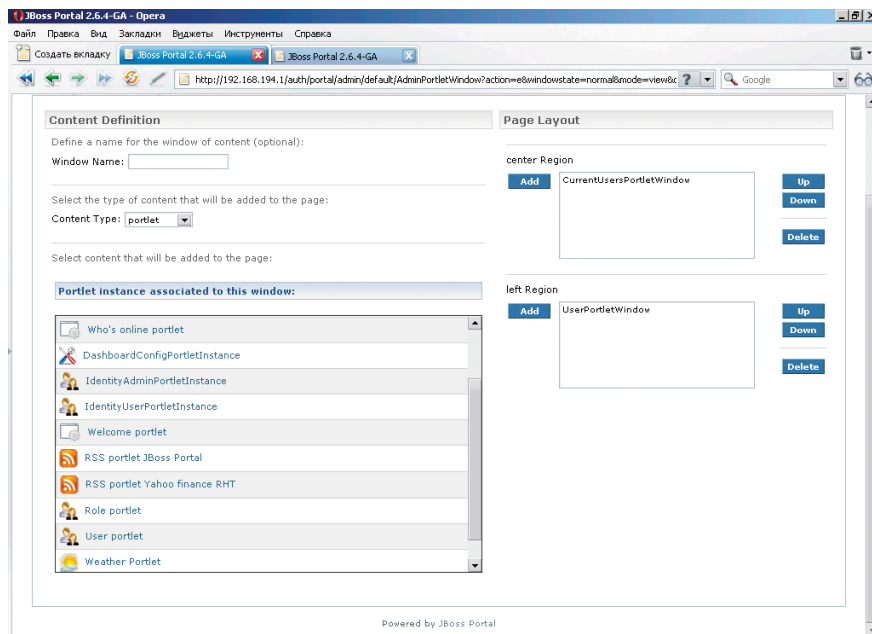


Рисунок 8. Размещаем на страницу портлеты

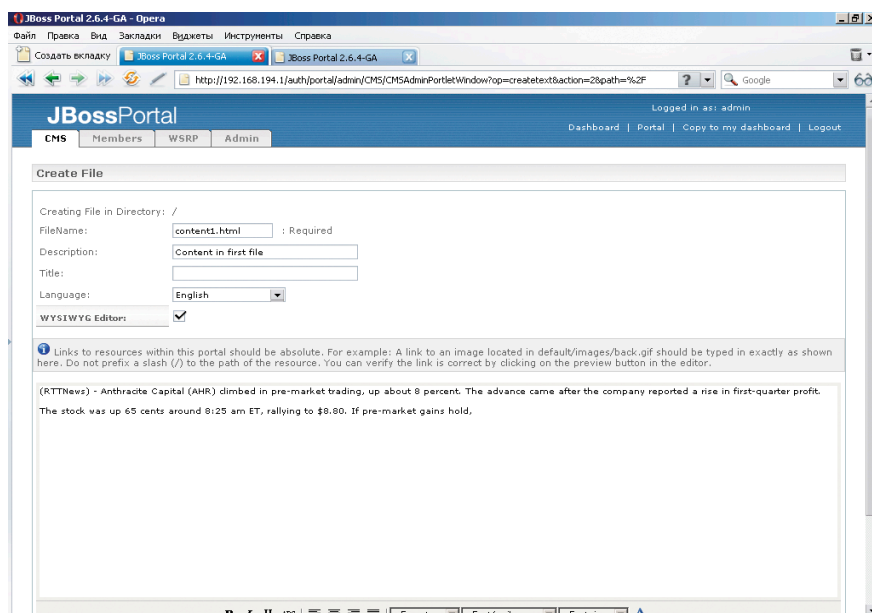


Рисунок 9. Работаем в CMS-редакторе

зывает, что страница не найдена. Впрочем, это еще впереди.

Переходим к наполнению страниц. На главную давайте поместим несколько стандартных портлетов из состава дистрибутива, вторую страницу сделаем композитной из нескольких HTML-страниц, а на остальные поместим свои собственные портлеты: один сторонний – GooglePortlet и один, который делает выборку из MySQL-базилища.

Мы находимся во вкладке «Portals → myPortal portal → Page1 page layout». Мы можем поместить на страницу предустановленные портлеты (Content

Type: «Portlets»), так называемые виджеты (Content Type: «widget/netvibes»), виджеты с сайта google.com (Content Type: «widget/google») или непосредственно скомпоновать из различных HTML-документов (Content Type: «cms»). Разместим портлеты Who's Online и User на данной странице (см **рис. 8**).

Для второй страницы предварительно нужно создать несколько HTML-файлов. Для этого переходим в раздел CMS и создаем там файлы: content1.html, content2.html, content3.html. Нажимаем на кнопку «Select Action...» и выбираем «Create File» (см. **рис. 9**).

Во встроенном HTML-редакторе придумываем содержимое, редактируем и украшаем, а затем публикуем, нажав на «Create» (см. рис. 10).

Аналогично для двух других файлов. В итоге должно получиться оформление, как на рис. 11.

Добавление собственных портлетов

Портлетов, идущих в комплекте дистрибутивной поставки, не так уж много. Поэтому возникает законный вопрос, а что нужно сделать, чтобы разместить портлет стороннего производителя? Как мы помним, JBP работает на Java, и спецификация JSR168 для него распространяется в полной мере. Поэтому те портлеты, которые спроектированы в полном соответствии с указанной спецификацией, должны запускаться и на нашем портале.

Поисковые службы для любого портала, как мне кажется, являются одной из главных составных частей. Закономерно, что был выбран механизм поиска компании Google – GooglePortlet. Если вы помните, то в 2002 году они представили свой сервис, известный как Google SOAP Search API. Разработчикам предлагалось, однажды зарегистрировавшись на сайте Google, получить доступ к весьма удобному интерфейсу поисковой машины.

Забираем архив с GooglePortlet по адресу [5], затем архив с jar-файлом от Google SOAP API по ссылке [6]. Чтобы портлет заработал, следует разместить googleapi.jar в путь переменной classpath. Либо поместить его в сам портлет, в директорию lib.

Но и это пока не все, следующий шаг – это размещение ключевого файла от поискового сервиса Google. На данный момент генерация новых ключей приостановлена, поэтому воспользуемся уже существующим ключом (если он у вас есть) либо используем ключ NC8jGG8c+SYzP3IuclPD+a0IFjA87Yd. Его следует поместить в файл portlet.xml в раздел <init-param>.

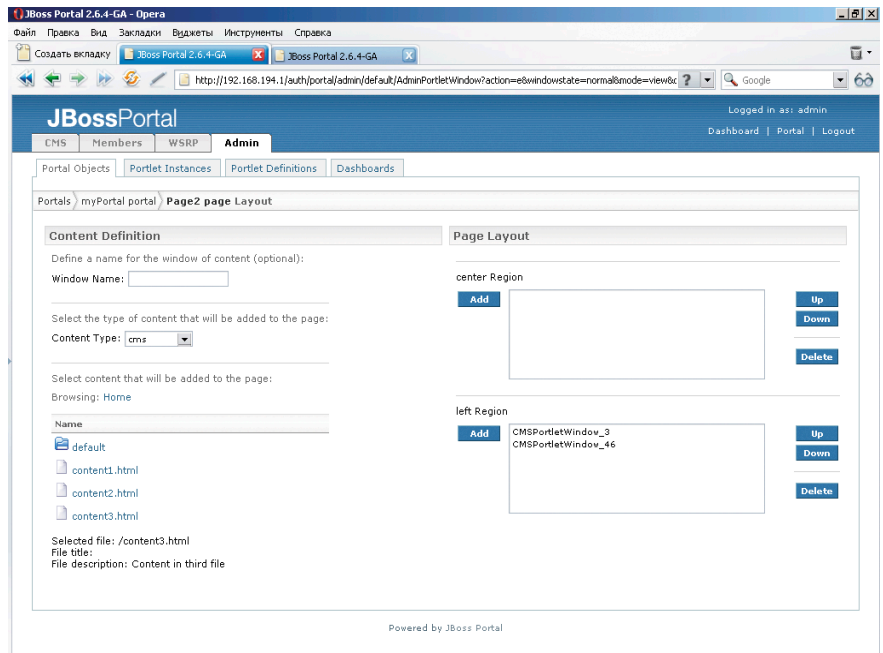


Рисунок 10. В итоге структура очередной страницы портала состоит из нескольких HTML-страниц

```
<portlet-class>com.plumtree.portlet.portlets. GooglePortlet</portlet-class>
<init-param>
  <name>key</name>
  <value>NC8jGG8c+SYzP3IuclPD+a0IFjA87Yd</value>
</init-param>
```

Размещаем портлет в JBP, скопировав его в директорию \$JBoss_HOME/server/default/deploy.

И затем аналогичным образом, так же как мы поступили с идущими в комплекте стандартными портлетами, разместим его на третьей странице. Полюбоваться поиском можно, перейдя на раздел «Страница3» портала и набрав, например, «BMW AG» (см. рис. 12).

Для более серьезных решений, когда сайт давно присутствует в Глобальной паутине и проиндексирован поисковиком, можно сделать контекстный поиск.

Например, сформировав запрос вида «Слово_для_поиска + Название_Своей_Компании», где «Слово_для_поиска» берется из собственного портлета, а «Название_Своей_компании» подставляется в строку запроса из этого же портлета.

С другими особенностями Google API можно ознакомиться на сайте GoogleDuel [7].

И наконец, пора разместить свой собственный портлет на последней странице портала. Он несложен, производит выборку новостных событий из базы и выводит ее результат в тело портлета. Воспользуемся примером портлета «Hello, World», правда, добавив несколько конструкций для доступа в MySQL-хранилище. Этот пример можно забрать со страницы JBoss Labs [8].

Оригинальный портлет выглядит следующим образом:

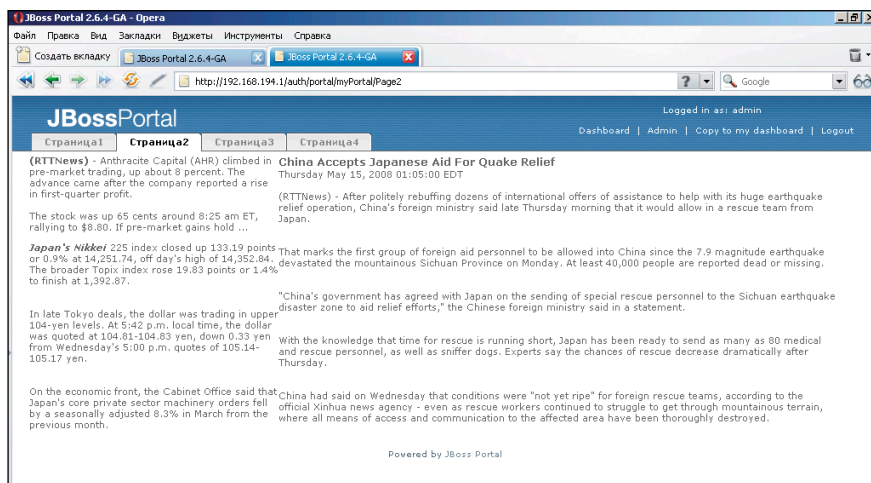


Рисунок 11. А так выглядит «составная» страница в окне браузера пользователя

```

package org.jboss.portlet.hello;

import javax.portlet.GenericPortlet;
import javax.portlet.PortletException;
import javax.portlet.RenderRequest;
import javax.portlet.RenderResponse;
import javax.portlet.UnavailableException;
import java.io.IOException;
import java.io.PrintWriter;

public class HelloWorldPortlet extends GenericPortlet
{
    protected void doView(RenderRequest rRequest,
        RenderResponse rResponse) throws PortletException,
        IOException, UnavailableException
    {
        rResponse.setContentType("text/html");
        PrintWriter writer = rResponse.getWriter();
        writer.write("Hello World!");
        writer.close();
    }
}

```

Модифицированный портлет:

```

1 package org.jboss.portlet.samplesql;

2 import javax.portlet.GenericPortlet;
3 import javax.portlet.PortletException;
4 import javax.portlet.RenderRequest;
5 import javax.portlet.RenderResponse;
6 import javax.portlet.UnavailableException;
7 import java.io.IOException;
8 import java.io.PrintWriter;

9 import java.sql.*;
10 import java.util.Properties;

11 public class SampleSQLPortlet extends GenericPortlet
12 {

13     Connection conn = null;
14     String url = "jdbc:mysql://localhost:3306/";
15     String userName = "jbossuser";
16     String password = "";

17     public void init()
18     {

```

```

19     }

20     protected void doView(RenderRequest rRequest,
        RenderResponse rResponse) throws PortletException,
        IOException, UnavailableException
21     {
22         rResponse.setContentType("text/html");
23         PrintWriter writer = rResponse.getWriter();

24         try
25         {
26             Class.forName(
                ("com.mysql.jdbc.Driver").newInstance ( ));
27             conn = DriverManager.getConnection (url,
                userName, password);
28             Statement s = conn.createStatement ( );

29             s.executeQuery ("SELECT * from
                html_data.news order by id");
30             ResultSet rs = s.getResultSet ( );
31             while (rs.next ( ))
32             {
33                 int uid = rs.getInt (1);
34                 String name = rs.getString(2);
35                 String name3 = name.replaceAll("\n",
                    "<br>");
36                 writer.println("<b>NewsID: " + uid +
                    "</b>, content: " + name3 +
                    "<br><br>");
37                 writer.flush();
38             }
39             rs.close ( );
40             s.close ( );

41         } catch (Exception e) { System.err.println
            ("Cannot connect to server" + e);

42         }

43         finally
44         {
45             if (conn != null)
46             {
47                 try
48                 {
49                     conn.close ( );
50                 } catch (Exception e) { /* ignore close
                    errors */ }

51             }

```

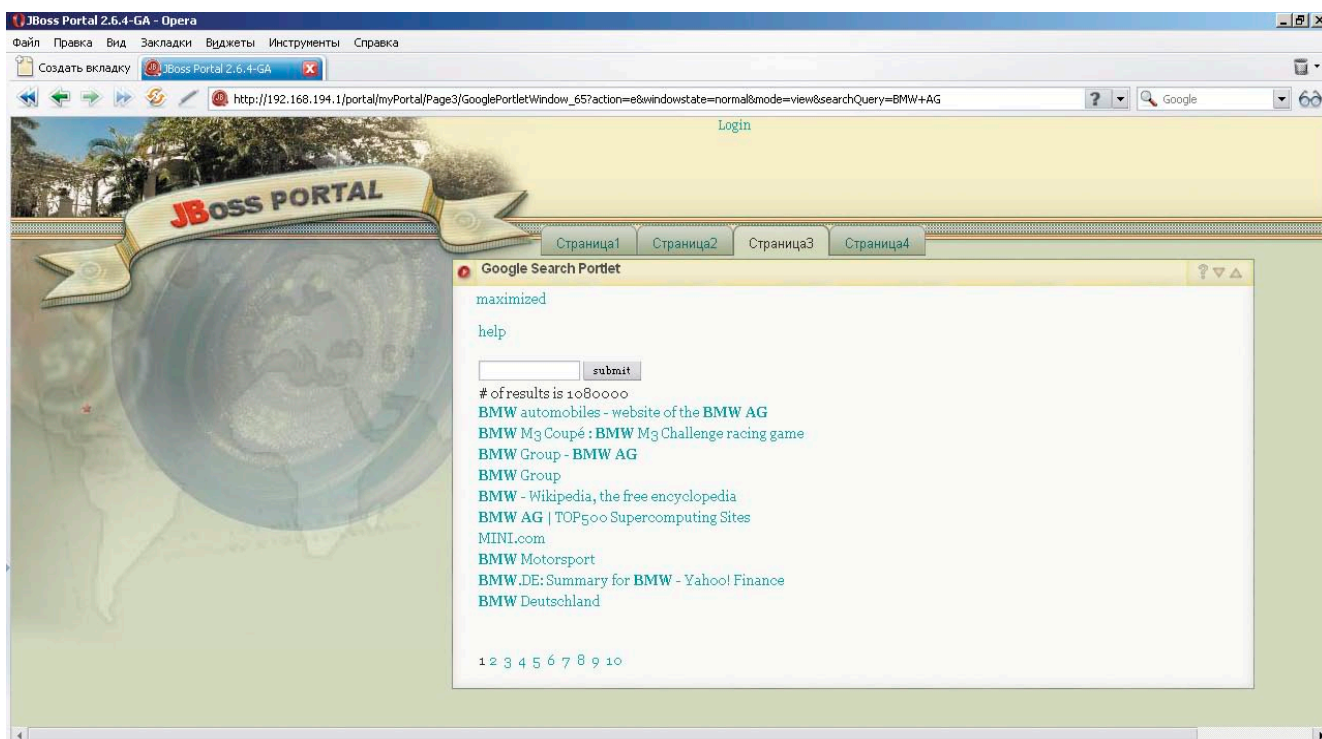


Рисунок 12. Поисковый портлет от Google в действии

```

52     writer.println("<br><b>(TM) BoldText</b>");
53     writer.close();
54 }
55 }

56 }

```

В данном примере мы расширили функционал стандартной функции портлета doView(), добавив обработку запроса к MySQL-хранилищу (строки 24-41) по адресу localhost:3306, под именем пользователя jbossuser и с пустым паролем (строки 13-16). Выборка осуществлена запросом (строка 29):

```
"SELECT * from html_data.news order by id"
```

Так как портлет – это веб-приложение, то представление в нем уже является HTML-страницей. Поэтому весь тот текст, что мы выводим из базы, нужно выводить как HTML. В этой связи все переводы строк заменены на их HTML-эквиваленты (строки 34-35), плюс мы еще выделили жирным номер новости (строка 36).

Не забудьте поменять название портлета в файле сборки build.xml с «helloworldportlet» на «samplesqlportlet», а также в файлах-ресурсах в директории resources/samplesqlportlet-war/WEB-INF/.

Сборка портлета относительно проста:

```
ant clean; ant
```

Готовый портлет можно найти в директории SampleSQLPortlet/output/lib. Его следует переписать в директорию горячего разворачивания портала либо сделать разворачивание портлета в автоматическом режиме. Для этого в build.xml вносим конструкцию копирования в нужное нам место:

```

<target name="deploy" depends="clover-yes, clover-no">
  <javac srcdir="${src.dir}"
        destdir="${classes.dir}"
        classpath="${libs}"
        debug="off"
        optimize="on"
        deprecation="on"
        compiler="${compiler}">
    <include name="main/org/jboss/portlet/**/*.java"/>
  </javac>

  <!-- portal-hello-lib.jar -->
  <jar jarfile="${build.lib}/samplesqlportlet-lib.jar">
    <fileset dir="${classes.dir}" />
  </jar>

  <mkdir dir="${build.resources}/samplesqlportlet-war/ WEB-INF/lib/" />
  <copy file="${build.lib}/samplesqlportlet-lib.jar"
        todir="${build.resources}/samplesqlportlet-war/ WEB-INF/lib/" />

  <!-- portal-hello.war -->
  <jar jarfile="${build.lib}/samplesqlportlet.war">
    <fileset dir="${build.resources}/ samplesqlportlet-war/" />
  </jar>

  <copy todir="/home/jboss/jboss-portal/server/ default/deploy" file="${build.lib}/ samplesqlportlet.war" />

</target>

```

Тогда каждая новая сборка портлета будет автоматиче-

ски скопирована и в директорию разворачивания портала и автоматически им развернута.

```

17:21:07,200 INFO [TomcatDeployer] deploy, ctxPath=/samplesqlportlet, warUrl=.../tmp/deploy/tmp41874samplesqlportlet-exp.war/
17:21:07,262 INFO [PortletAppDeployment] Parsing /samplesqlportlet/jboss-portlet.xml
17:21:07,285 INFO [PortletAppDeployment] These instances have been found in -object.xml, you should put them in the file /home/jboss/jboss-portal-2.6.4/server/default/.../tmp/deploy/tmp41874samplesqlportlet-exp.war/WEB-INF/portlet-instances.xml
17:21:07,285 INFO [PortletAppDeployment] <?xml version="1.0" encoding="utf-8" standalone="yes"?>
<deployments>
<deployment>
<instance>
<instance-id>SampleSQLPortletInstance</instance-id>
<portlet-ref>SampleSQLPortlet</portlet-ref>
</instance>
</deployment>
</deployments>

```

При этом в портале, в административном режиме, вы увидите, что появился наш портлет под именем SampleSQL PortletInstance (см. **рис. 13**).

Его-то и размещаем на четвертой странице. Выглядит, на мой взгляд, любопытно (см. **рис. 14**).

Смена визуальной цветовой «одежки»

Конечно, было бы интересно сделать сайт визуально различимым, хотя бы в цветовой гамме. Для этого в стандартной установке присутствуют следующие темы: industrial, renaissance, maple, nphalanx и mission-critical. В частности, industrial выглядит вот так.

Вы обратили внимание, что в примерах используется отнюдь не стандартная тема. Все правильно, тема называется «havana-affair» и может быть загружена с сайта JBoss, подраздел portletswap: <http://www.jboss.org/portletswap/downloads/layouts>.

Давайте рассмотрим, что же из себя представляет тема. Распаковав файл havana_affair.zip, мы видим, что содержимое темы (графические файлы и CSS-раскладка) хранятся в директории havana_affair.war. Тема, как и портлеты, может быть развернута в реальном режиме – также как и портлеты, данную директорию нужно переписать в \$JBoss_HOME/server/default/deploy.

Посмотрим на CSS-раскладку темы – файл «portal_style.css». За описание порталных страниц отвечают теги: #body, #spacer, #header-container, #header, #logoName. Так, тэг #body выглядит следующим образом:

```

#body {
    background-image: url(images/portal_background.gif);
    background-repeat: repeat-x;
    background-color: #d5d9be;
    margin: 0px;
    padding: 0px;
    font-family: Georgia, "Times New Roman", Times, serif;
    font-size: 12px;
    color: #353634;
}

```

Что это означает – страницы будут «залиты» фоновым рисунком из файла portal_background.gif. Причем если рисунок представляет собой полосу, то фон страницы будет составлен из множества таких полосок – за это отвечает параметр «background-repeat: repeat-x». Цвет страницы в нашем примере: #d5d9be (в hex-нотации R-G-B). Ис-

пользуется шрифт, указанный в параметре font-family с размером в 12 пунктов. Его цвет – #353634.

Как видите, с дизайнерской точки зрения понятно. Первым делом после установки портала хочется поменять логотип компании – в тэге #logoName за это отвечает параметр background-image. Осталось только не запутаться с другими тэгами, отвечающими за более тонкую раскладку, такими как, например, #regionA, #regionB, #regionC, UL#tabsHeader. Они подробно описаны в документации к portalу, поэтому если вы хотите сделать «лицо» компании действительно эксклюзивным, то прочитайте раздел «24. Layouts and Themes» из JBossPortal Reference Guide.

Стоит также отметить, что за представление портлетов отвечают тэги вида .portlet-название. Например, .portlet-container, .portlet-titlebar-title, .portlet-titlebar-decoration и другие.

Так, тэг .portlet-body управляет цветовой гаммой портлета

```
.portlet-body {
    background-color: #fafaf9;
}
```

В свете вышеизложенного можно и нужно попросить вашего дизайнера поработать над вашим собственным стилем. Знание CSS, разметки страницы и артистический вкус – и вас запомнят. Темы, как и портлеты, разворачиваются в «горячем» режиме – копированием в директорию deploy.

Пару слов об управляемости портлетов. Как мы уже показали выше, портлеты разворачиваются в реальном времени – нет смысла останавливать портал, копировать единичные портлеты и запускать заново портал. Если вы составляете свой собственный портлет, то внимательно следите за тем, как вы составляете параметры (файлы-ресурсы должны иметь корректные параметры). В противном случае портлет не развернется.

Мы рассмотрели одну из самых простых реализаций портлета – SampleSQLPortlet. От него не требуется какой-то сложной динамики. Главная его задача – выводить информацию из локальной базы (см. рис. 15).

Добавив более сложную бизнес-логику, мы можем сделать его интеллектуальнее, например, выводить информацию за последние несколько

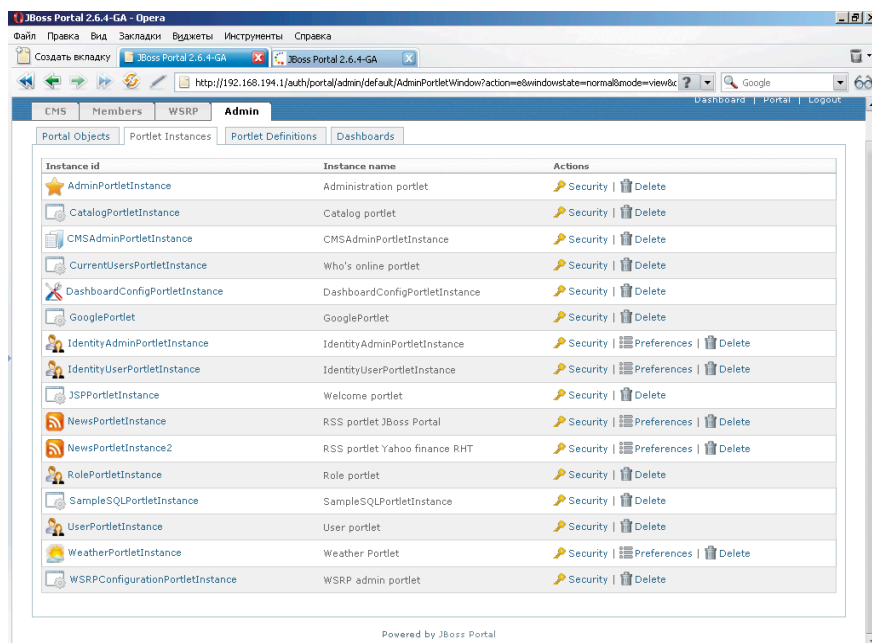


Рисунок 13. В списке портлетов появился и наш

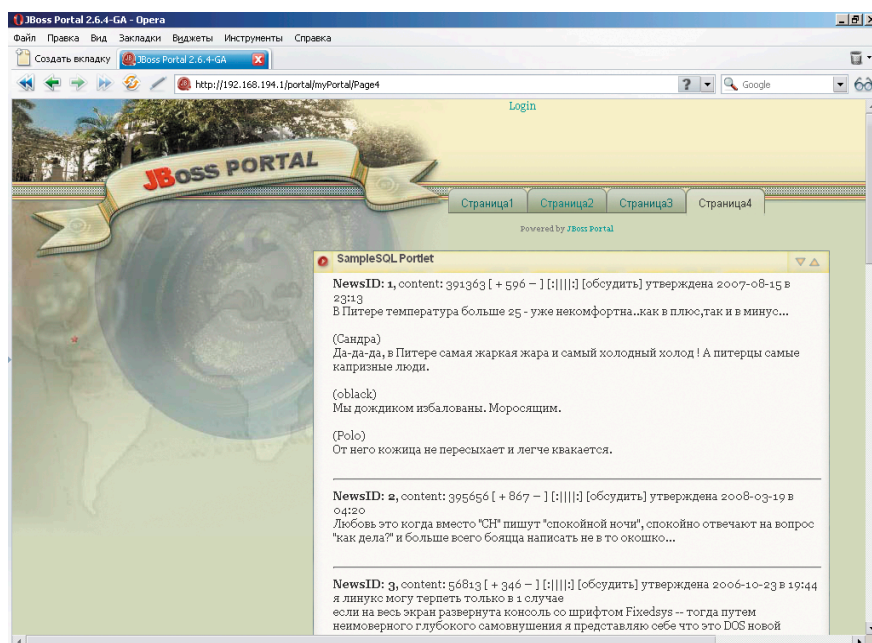


Рисунок 14. Результат выборки MySQL-базы в теле портлета

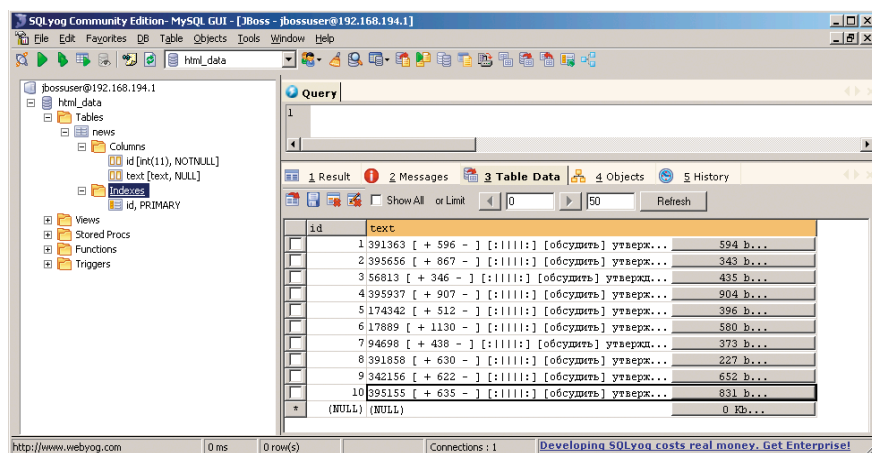


Рисунок 15. Информация, что будет выводить SampleSQLPortlet

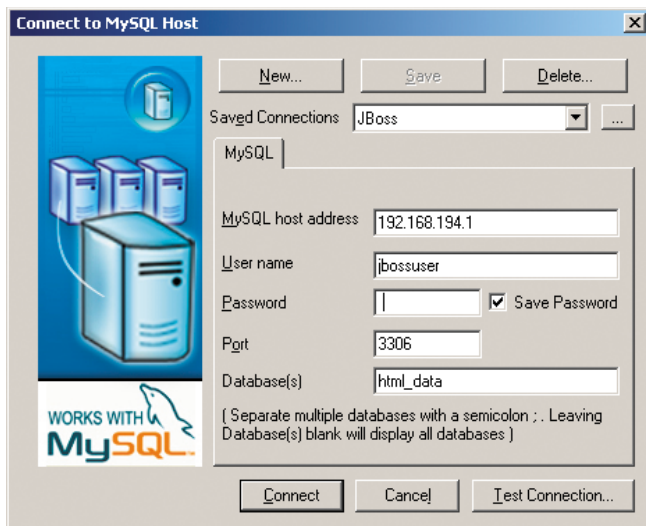


Рисунок 16. К MySQL-базе можно подключаться с помощью, например, SQLyog

дней. Отмечу также, что обновление содержимого данной базы (html.data) не представляет из себя что-то архисложное (см. рис. 16, 17).

В качестве инструмента обновления мы использовали SQLyog, но также можно использовать и любое другое решение, в зависимости от квалификации оператора и поставленных задач.

Где взять дополнительные портлеты? Например, на сайте JBoss в разделе portletswap. Также можно обратиться к ресурсам <https://gems.dev.java.net>, <http://www.adenin.com/Intranet-Suite/Enterprise-Portal.aspx>, <http://www.syncex.com>, <http://portlet-opensrc.sourceforge.net>.

Цена вопроса

При проектировании портала стоит помнить, что ваш выбор не ограничивается JBoss Portal. Вы можете взять, например, конкурентные продукты – Liferay, uPortal, GridSphere (они все принадлежат к классу Open Source) и попробовать построить порталное решение с их помощью. С финансовой точки зрения вы оплачиваете только свои собственные издержки, и, как правило, они будут значительно ниже покупки коммерческих решений.

Тем компаниям, что принадлежат к сегменту малого и среднего бизнеса (SMB) и ценят время на разворачивание, но не видят возможности пойти на крупные капиталъ-

ные вложения, стоит посоветовать подписаться на техническую поддержку от JBoss. Понятно, что за советы у нас не очень любят платить, но, с другой стороны – ничего бесплатного на дороге не валяется. И если компания свободна во временных рамках, то это классический пример, когда оплата за продукт неявным образом переходит на оплату труда своих же сотрудников.

По мере роста масштабов производства и, соответственно, потоков информации закидываться на стоимости продукта и его внедрении уже не имеет смысла. В этом случае коммерческие решения выливаются в следующие примерные суммы:

- **IBM WebSphere Portal Server** – 51500 долларов за процессор;
- **IBM WebSphere Portal Express** – 2300 долларов за пакет на 20 пользователей или лицензия за каждый CPU до 39999 долларов;
- **Oracle AS Portal** – лицензия за одного пользователя – 200 долларов или лицензия за каждый CPU – 10000 долларов;
- **Oracle AS Portal Standard Edition One** – лицензия за одного пользователя – 149 долларов или лицензия за каждый CPU – 4995 долларов.

На фоне курсов для JBoss Portal 2250 евро выглядит впечатляюще. Сравнение, конечно же, не совсем корректное, но позволяет судить о масштабах цен.

И в качестве заключения процитирую Юлия Матеевского (директора аналитического департамента Dresdner Kleinwort Wasserstein): «Сейчас в России многие компании малого бизнеса предпочитают не вкладывать средства в развитие внутренних IT-решений, ограничиваясь корпоративным сайтом для сотрудников. Как правило, такой подход не приносит компании особой пользы и не влияет на эффективность ее деятельности. Однако при дальнейшем расширении оргструктуры все равно, рано или поздно, приходится использовать хотя бы минимальный пакет IT-услуг. Иначе эта самая расширенная оргструктура превратится в колосса на глиняных ногах».

1. http://www.kck.ru/kcksite/kckmain2008.nsf/pages/spheres_all.html.
2. <http://www.microsoft.com/Rus/Government/Solutions/portal/architecture/default.mspx>.
3. <http://downloads.sourceforge.net/jboss/jboss-portal-2.6.4-bundled.zip>.
4. <http://dev.mysql.com/downloads/connector/j/5.1.html>.
5. <http://prdownloads.sourceforge.net/portlet-opensrc/GooglePortlet.zip?download>.
6. <http://debian.fmi.uni-sofia.bg/~hiena/googleapi.zip>.
7. <http://www.googleduel.com/apiexample.php>.
8. http://labs.jboss.com/portal/portletswap/portlets_tutorial.html.
9. http://en.wikipedia.org/wiki/JSR_168.
10. http://en.wikipedia.org/wiki/Enterprise_portal.

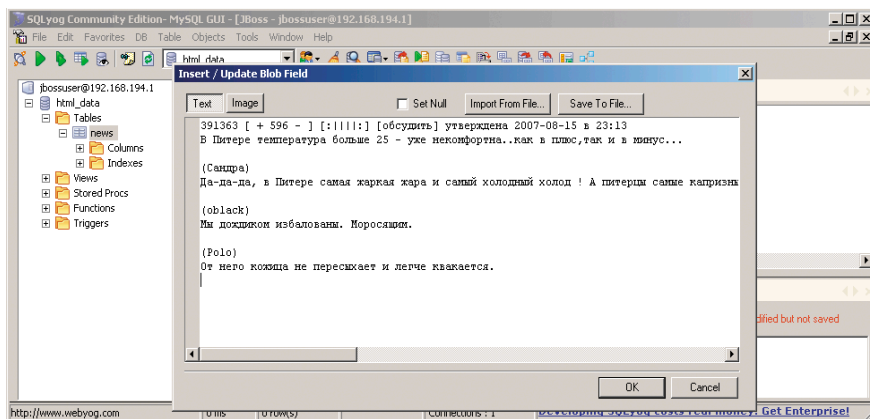


Рисунок 17. Заполнять новыми значениями базу легче простого