

Описание и модернизация приложения визуализации данных inGraph

Козлов Владимир Олегович

БНТУ, Минск, Республика Беларусь, groprinosin@gmail.com

Введение

Данная работа описывает процесс подготовки окружающей среды и развёртывания фреймворка Agavi. Так же в работе будет описан процесс подключения библиотеки Doctrine, которая необходима для создания моделей БД, для поддержания концепции Model View Controller (MVC). Так же в ходе работы будут представлены примеры модернизации проекта с использованием модульной концепции фреймворка Agavi. В ходе работы будет рассмотрено приложение inGraph, реализованное на основе фреймворка Agavi. Данное приложение позволяет отображать визуальную информацию в виде графиков.

Подготовка к работе

Разрабатываемое приложение базируется на CentOS 6.0. После установки операционной системы на виртуальную машины для создания полноценного сервера необходимо установить apache, php и mysql. Все действия выполняются от пользователя root. Для установки системы необходимо выполнить следующие шаги.

Шаг 1. Необходимо обновить систему. Это делается следующей командой:

```
# yum update
```

Шаг 2. Для удобной работы необходимо установить Midnight Commander:

```
# yum -y install mc
```

Шаг 3. Установка Apache, SSL:

```
# yum -y install httpd mod_ssl
```

Далее необходимо отредактировать конфигурационный файл apache. Для этого необходимо открыть в любимом редакторе файл `/etc/httpd/conf/httpd.conf`. В `ServerName` вносится адрес или имя сервера текущей копии системы. Прописывается Apache в автозапуск:

```
# chkconfig httpd on
```

Apache запускается командой:

```
# service httpd start
```

Для проверки работы сервера создаем тестовую страницу:

```
# echo '<h1>It Works!</h1>' > /var/www/html/index.html
```

Проверяем результат, открыв в браузере с другой машины IP-адрес сервера:

http://IP-адрес_сервера/

Если надпись It Works! появилась, значит сервер работает корректно.

Шаг 4. Установка PHP и его дополнительные компоненты. Для этого необходимо выполнить команду:

```
# yum -y install php php-common php-gd php-mysql php-xml php-mbstring
```

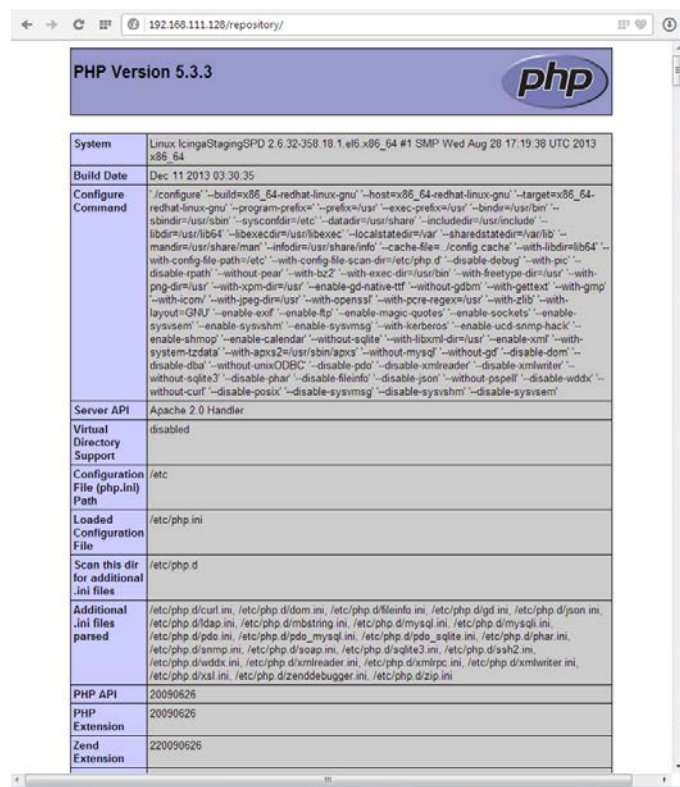
Далее необходимо перезапустить Apache:

```
# service httpd restart
```

Проверить работу PHP можно выполнив следующий скрипт:

```
# echo '<?php phpinfo(); ?>' > /var/www/html/inf.php
```

Результат работы данного скрипта проиллюстрирован на рисунке 1.



System	Linux fc16ngStagingSPD 2.6.32-358.18.1.el6.x86_64 #1 SMP Wed Aug 28 17:19:38 UTC 2013 x86_64
Build Date	Dec 11 2013 03:30:35
Configure Command	./configure '--build=x86_64-redhat-linux-gnu' '--host=x86_64-redhat-linux-gnu' '--target=x86_64-redhat-linux-gnu' '--program-prefix=' '--prefix=/usr' '--exec-prefix=/usr' '--bindir=/usr/bin' '--sbindir=/usr/sbin' '--sysconfdir=/etc' '--datadir=/usr/share' '--includedir=/usr/include' '--libdir=/usr/lib64' '--libexecdir=/usr/libexec' '--localstatedir=/var' '--sharedstatedir=/var/lib' '--mandir=/usr/share/man' '--infodir=/usr/share/info' '--cache-file=/config.cache' '--with-libdir=lib64' '--with-config-file-path=/etc' '--with-config-file=/etc/php.d' '--disable-debug' '--with-pc' '--disable-path' '--without-pear' '--with-ldap2' '--with-exec-dir=/usr/bin' '--with-freezetype-dir=/usr' '--with-png-dir=/usr' '--with-xpm-dir=/usr' '--enable-gd-native-ttf' '--without-gdgm' '--with-gettext' '--with-gmp' '--with-iconv' '--with-jpeg-dir=/usr' '--with-openssl' '--with-pcre-regex=/usr' '--with-zlib' '--with-layout=GNU' '--enable-exif' '--enable-ftp' '--enable-magic-quotes' '--enable-sockets' '--enable-sysvsem' '--enable-sysvshm' '--enable-sysvmsg' '--with-kerberos' '--enable-ucd-snmp-back' '--enable-ahmop' '--enable-calendar' '--without-sqlite' '--with-libxml-dir=/usr' '--enable-xml' '--with-system-tzdata' '--with-apxs2=/usr/sbin/apxs' '--without-mysql' '--without-gd' '--disable-dom' '--disable-dba' '--without-unixODBC' '--disable-pdo' '--disable-xmlreader' '--disable-xmlwriter' '--without-sqlite3' '--disable-phar' '--disable-fileinfo' '--disable-json' '--without-pspell' '--disable-wddx' '--without-curl' '--disable-posix' '--disable-sysmsg' '--disable-sysvshm' '--disable-sysvsem'
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/curl.ini, /etc/php.d/dom.ini, /etc/php.d/imap.ini, /etc/php.d/ldap.ini, /etc/php.d/mcrypt.ini, /etc/php.d/mysqli.ini, /etc/php.d/pdo.ini, /etc/php.d/pdo_mysql.ini, /etc/php.d/pdo_sqlite.ini, /etc/php.d/phar.ini, /etc/php.d/snmp.ini, /etc/php.d/soap.ini, /etc/php.d/sqlite3.ini, /etc/php.d/sh2.ini, /etc/php.d/wddx.ini, /etc/php.d/xmlreader.ini, /etc/php.d/xmlrpc.ini, /etc/php.d/xmlwriter.ini, /etc/php.d/xsl.ini, /etc/php.d/ZendDebugger.ini, /etc/php.d/zip.ini
PHP API	20090626
PHP Extension	20090626
Zend Extension	220090626

Рисунок 1 – Пример работы PHP

Шаг 5. Установка MySQL. Для установки MySQL выполняем команду:

```
# yum -y install mysql mysql-server
```

Добавляем MySQL в автозапуск командой:

```
# chkconfig mysqld on
```

MySQL запускается следующей командой:

```
# service mysqld start
```

Далее необходимо установить root пароль MySQL:

```
# mysqladmin -u root password `root`
```

После выполнения данных операций, получается полноценный сервер на базе CentOS готовый к работе. Для удобной работы с платформой CentOS из Windows необходимо установить RHPStorm и настроить проект на автоматическую синхронизацию с сервером. Для этого необходимо открыть вкладку Tools/Deployment/Configurations и указать директорию синхронизации на сервере. Пример данной настройки приведен на рисунке 2.

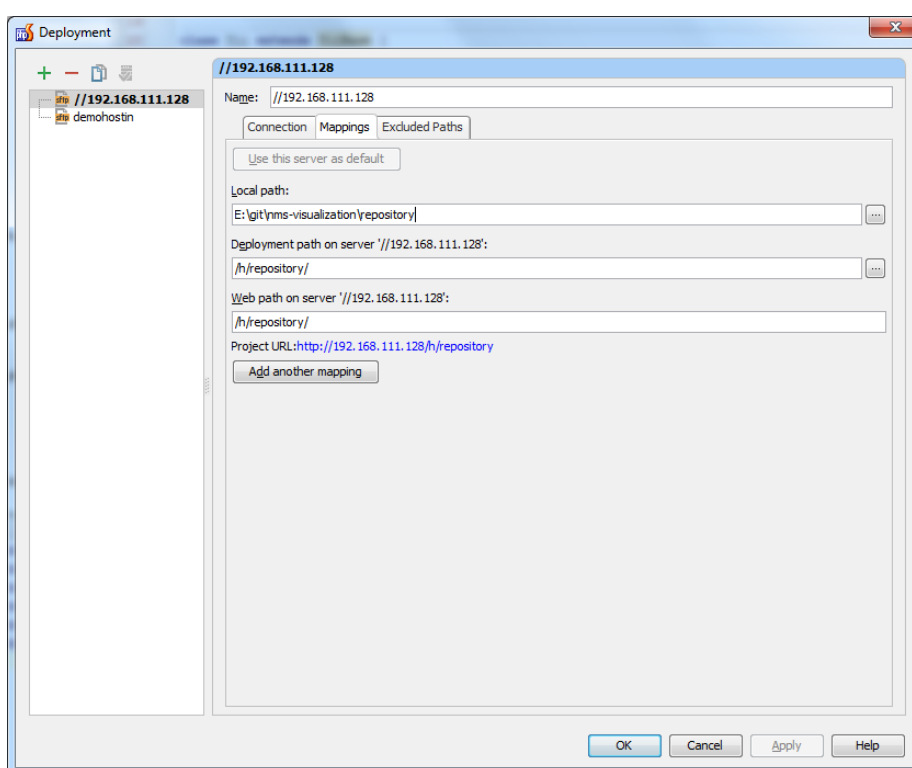


Рисунок 2 – Пример настройки синхронизации проекта

Установка Agavi Framework

Используя описанную выше настройку сервера, развернуть Agavi Framework достаточно просто. Для этого необходимо скачать пакет Agavi и с помощью RHP Storm загрузить его на сервер. Стандартный пакет Agavi имеет структуру, представленную на рисунке 3.

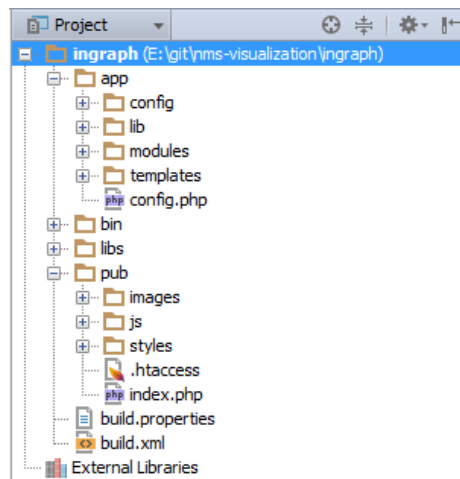


Рисунок 3 – Структура Agavi Framework

Директория app содержит исполняемые файлы, файлы конфигурации фреймворка, а так же файлы для формирования отображения страницы. Директория pub содержит элементы контроллера. В данной директории содержатся js-скрипты, изображения для сайта и CSS-файлы для настройки отображения. Директория libs содержит вспомогательные библиотеки. В нашем случае таковой является Doctrine.

Для установки Doctrine необходимо скачать архив Doctrine и распаковать его в директорию libs. Далее необходимо сгенерировать модели базы данных для того, что бы работать с базой данных repository по средствам классов php. Для генерации моделей необходимо выполнить скрипт, листинг которого представлен на рисунке 4.

```
1 <?php
2 // include main Doctrine class file
3 // change this per your system
4 include '/usr/local/ingraph/libs/Doctrine.php';
5 spl_autoload_register(array('Doctrine', 'autoload'));
6
7 // create Doctrine manager
8 $manager = Doctrine_Manager::getInstance();
9
10 // create database connection
11 $conn = Doctrine_Manager::connection('mysql://root:root@localhost/repository', 'doctrine');
12
13 // auto-generate models
14 Doctrine::generateModelsFromDb('models', array('doctrine'),
15 array('generateTableClasses' => true));
16 ?>
```

Рисунок 4 – Пример скрипта генерации моделей

После этого будут сгенерированы модели базы данных. Пример моделей приведён на рисунке 5.

```

49 abstract class BaseDevices extends Doctrine_Record
50 {
51     public function setTableDefinition()
52     {
53         $this->setTableName('devices');
54         $this->hasManyColumn('id_device', 'integer', 4, array('type' => 'integer', 'length' => 4, 'primary' => true, 'autoincrement' => true));
55         $this->hasManyColumn('mac', 'string', 23, array('type' => 'string', 'length' => 23, 'nullable' => true));
56         $this->hasManyColumn('ip', 'string', 45, array('type' => 'string', 'length' => 45));
57         $this->hasManyColumn('uptime', 'integer', 4, array('type' => 'integer', 'length' => 4));
58         $this->hasManyColumn('type', 'integer', 4, array('type' => 'integer', 'length' => 4));
59         $this->hasManyColumn('location', 'string', 255, array('type' => 'string', 'length' => 255));
60         $this->hasManyColumn('owner', 'integer', 4, array('type' => 'integer', 'length' => 4));
61         $this->hasManyColumn('start_date', 'timestamp', null, array('type' => 'timestamp'));
62         $this->hasManyColumn('status', 'integer', 4, array('type' => 'integer', 'length' => 4));
63         $this->hasManyColumn('uptime', 'integer', 4, array('type' => 'integer', 'length' => 4));
64         $this->hasManyColumn('contact', 'integer', 4, array('type' => 'integer', 'length' => 4));
65         $this->hasManyColumn('model', 'string', 50, array('type' => 'string', 'length' => 50));
66         $this->hasManyColumn('serial', 'string', 50, array('type' => 'string', 'length' => 50));
67         $this->hasManyColumn('unauthorized', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
68         $this->hasManyColumn('archive', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '0', 'nullable' => true));
69         $this->hasManyColumn('archiving_date', 'timestamp', null, array('type' => 'timestamp'));
70         $this->hasManyColumn('mac_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
71         $this->hasManyColumn('host_name_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
72         $this->hasManyColumn('ip_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
73         $this->hasManyColumn('type_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
74         $this->hasManyColumn('location_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
75         $this->hasManyColumn('owner_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
76         $this->hasManyColumn('start_date_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
77         $this->hasManyColumn('status_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
78         $this->hasManyColumn('parent_id_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
79         $this->hasManyColumn('uptime_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
80         $this->hasManyColumn('contact_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
81         $this->hasManyColumn('model_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
82         $this->hasManyColumn('serial_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
83         $this->hasManyColumn('unauthorized_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
84         $this->hasManyColumn('services_edit', 'integer', 1, array('type' => 'integer', 'length' => 1, 'default' => '1', 'nullable' => true));
85     }
86
87     public function setUp()
88     {
89         $this->hasMany('DevicesHasServices', array('local' => 'id_device',
90             'foreign' => 'device_id_device'));
91
92         $this->hasMany('Parents', array('local' => 'id_device',
93             'foreign' => 'id_parent_device'));
94     }
95 }

```

Рисунок 5 – Пример скрипта сгенерированной модели

Для подключение библиотеки Doctrine необходимо настроить скрипт конфигурации agavi database.xml (рисунок 6).

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <config xmlns="http://agavi.org/agavi/config/global/version/1.0"
3     xmlns:db="http://agavi.org/agavi/config/parts/databases/1.0">
4
5     <db:configuration>
6         <db:repository default="repository">
7
8             <db:database xmlns="http://agavi.org/agavi/config/parts/databases/1.0" name="repository" class="AgaviDoctrineDatabase">
9                 <db:parameter name="dsn">mysql://root:root@localhost/repository/</db:parameter>
10
11                 <db:parameter name="prefix"></db:parameter>
12                 <db:parameter name="charset">utf8</db:parameter>
13                 <db:parameter name="data_format">YYYY-MM-DD HH24:MI:SS]]&gt;&lt;/db:parameter&gt;
14                 &lt;db:parameter name="manager_attributes"&gt;&lt;/db:parameter&gt;
15                 &lt;db:parameter name="Doctrine_Core::ATTR_MERGE_LOADING"&gt;CONSERVATIVE&lt;/db:parameter&gt;
16             &lt;/db:database&gt;
17
18             &lt;db:parameter name="use_retained"&gt;true&lt;/db:parameter&gt;
19
20             &lt;db:parameter name="load_models"&gt;
21                 &lt;db:parameter&gt;@core.lib_dir/doctrine/generated/&lt;/db:parameter&gt;
22                 &lt;db:parameter&gt;@core.lib_dir/doctrine/&lt;/db:parameter&gt;
23             &lt;/db:parameter&gt;
24
25             &lt;db:parameter name="caching"&gt;
26                 &lt;db:parameter name="enabled"&gt;false&lt;/db:parameter&gt;
27
28                 &lt;db:parameter name="driver"&gt;apcc&lt;/db:parameter&gt;
29                 &lt;db:parameter name="use_query_cache"&gt;true&lt;/db:parameter&gt;
30             &lt;/db:parameter&gt;
31
32         &lt;/db:repository&gt;
33     &lt;/db:configuration&gt;
34
35 &lt;/db&gt;
36
37 &lt;/config&gt;
</pre>
</div>
<div data-bbox="199 751 873 769" data-label="Caption">
<p><b>Рисунок 6 – Конфигурационный файл настройки подключения Doctrine</b></p>
</div>
<div data-bbox="138 782 940 824" data-label="Text">
<p>Выполнив эти действия, получим подключенную библиотеку Doctrine для работы с базой данных и модели классов, соответствующих таблицам базы данных.</p>
</div>
```

Реализация web-приложения

Для реализации поставленной задачи, прежде всего был установлен и настроен модуль Flot. Так же для данного модуля были реализованы следующие вспомогательные функциональные части:

- jquery.flot.description;
- jquery.flot.offset;
- jquery.flot.thresholds;
- jquery.flot.highlight.

Рассмотрим более подробно реализованные модули.

Jquery.flot.description. Данный модуль необходим для отображения вспомогательной информации на графике. Данный модуль работает с объектом Canvas и прорисовывает на нём буквы (именно прорисовывает, а не выводит текст). Пример данного вспомогательного модуля приведён на рисунке 7.

```
168 var total = 0;
169 var len = str.length;
170 var mag = size / 25.0;
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

Рисунок 7 – Пример исходного кода вспомогательного модуля

Значения для вывода берутся из json-объекта, который присылает provider/values.

Jquery.flot.offset. Данный модуль предназначен для прорисовки сопряжённых графиков устройств. Данные о сопряжённом графике приходят в отдельном объекте. Прорисовка происходит путём смещения исходного положения canvas на 10% относительно текущей позиции.

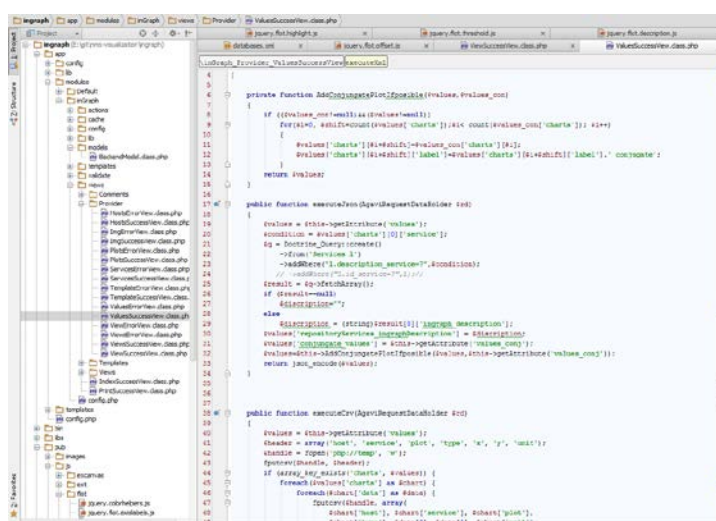
Jquery.flot.thresholds. Данный модуль отображает пороговые значения для графиков.

Jquery.flot.highlight. Данный модуль прорисовывает подсветку точек. При наведение на точку, она выделяется и во всплывающем окне высвечивается значение по осям x и y.

Модуль подключаются путем добавления исполняемой функции в виде хука на определенное событие. Например:

```
plot.hooks.draw.push(ReturnOffset);
```

Так же был изменён модуль формирования данных json-объекта. Данный модуль получает данные из базы данных repository и ingraph. Пример скрипта для формирования данных представлен на рисунке 8



```
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

Рисунок 8 – Пример скрипта формирование ответа в виде json-объекта

Тестирование

Локальное тестирование проходило с использованием браузеров Орега и InternetExplorer по методу белого ящика. Это позволило устранить значительную часть неисправностей и неточностей приложения. Значительная часть тестируемой информация была получена в виде json-объектов. Пример отладочной информации приведён на рисунке 9.

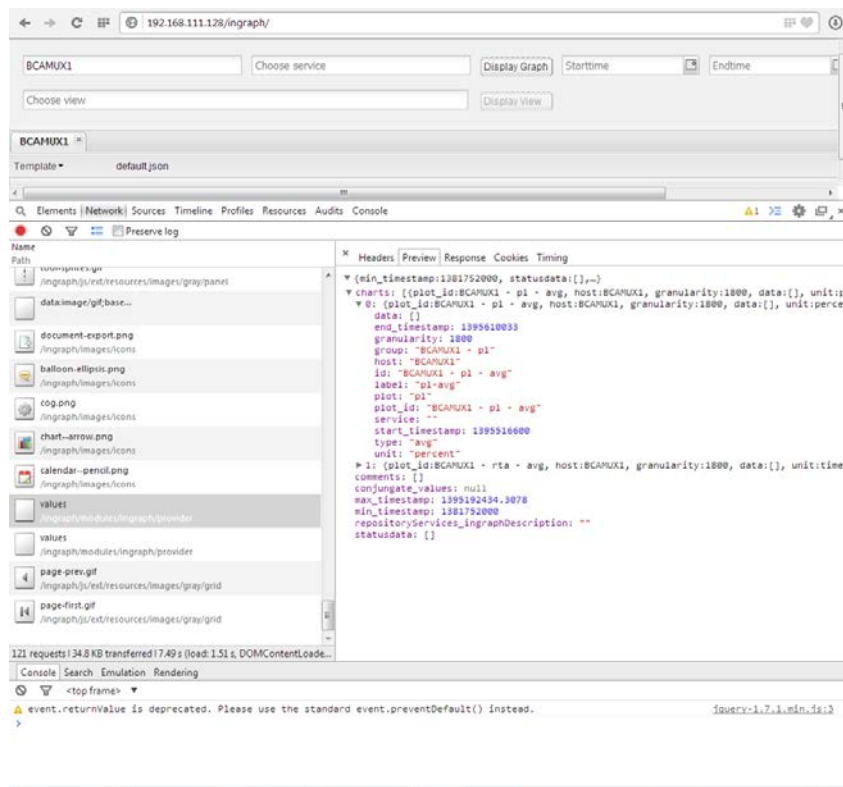


Рисунок 9 – пример отображения тестовой информации

В результате тестирования были найдены следующие недочёт – не отображается график в Internet Explorer v 8. Это обусловлено тем, что IE 8 не поддерживает элемент canvas. Решить эту проблему можно путём подключения к проекту библиотеки excanvas.js.

Пример работы готового приложения можно увидеть на рисунках 10-11.

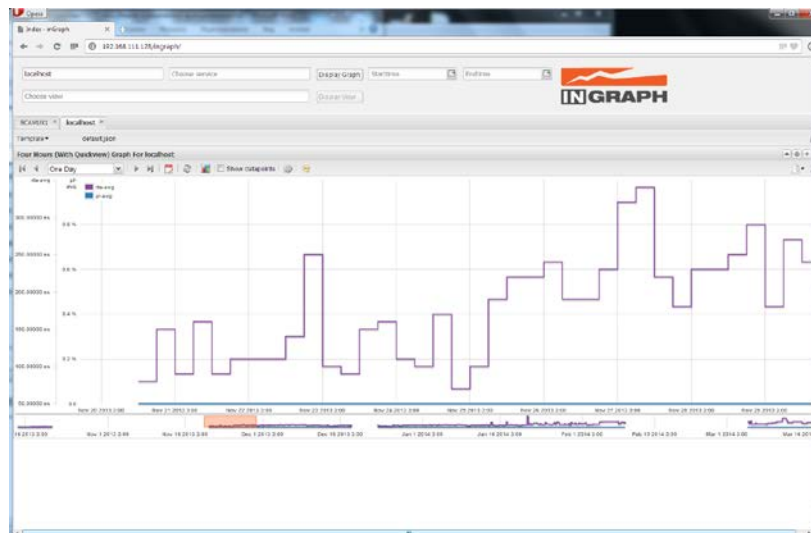


Рисунок 14 – Пример работы приложения

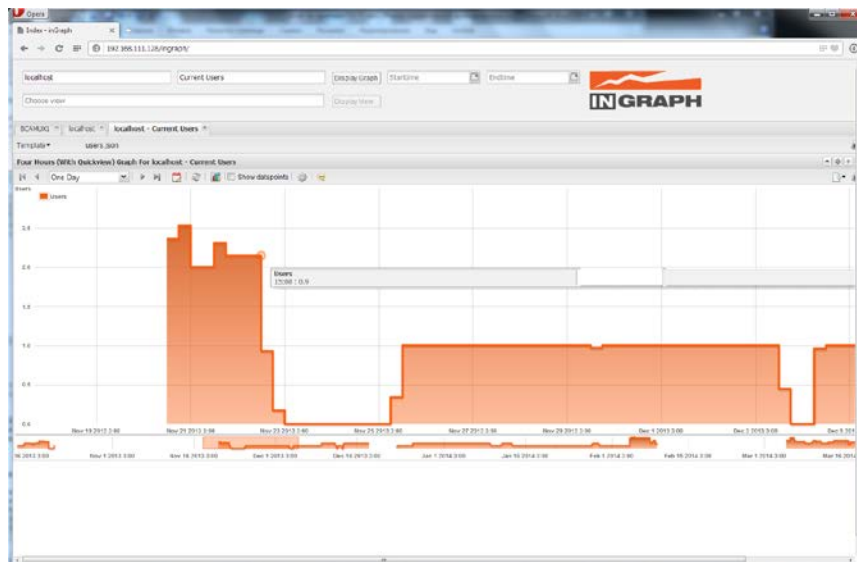


Рисунок 15 – Пример работы приложения

Вывод

В ходе работы было модернизировано web-приложение inGraph, которое является вспомогательным модулем для системы мониторинга компьютерных систем и сетей Icinga. Данное приложение прошло успешное тестирование. Приложение показало устойчивую и корректную работу. Для тестирования приложения с псевдозначениями были реализованы скрипты для генерации случайных значений.