



QlikView-OpenCPU Demo

OpenCPU and QlikView Demo Configuration

spa

June 2013

qlikview.com



Contact Information

Name of Company	Qlik Technologies, Inc.
Address	Global Headquarters 150 N. Radnor Chester Road Suite E220 Radnor, PA 19087
Telephone Number	+1 (888) 828-9768
Fax Number	+1 (610) 975-5987
Primary Contact	Samuel Pena
Designation	Solution Architect
Email	samuel.pena@qlikview.com
Date	June 2013

Installation and Setup

This brief document's purpose is to enable you to integrate R to the Cluster Analysis.qvw on demand using a cloud solution. More specifically, this paper will demonstrate how you can send data to and from other systems using the web as the pipeline. In this example, the Cluster Analysis.qvw will send data to Open CPU, Open CPU will process the data and send the data back to Cluster Analysis on demand. Open CPU is an initiative from Jeroem Ooms and has been implemented by Stat/Dev consulting in the Netherlands.

These are steps you will go through:

1. Add a R script to the Open CPU store
2. Deploy the Cluster Analysis.qvw to QlikView server
3. Install & Configure the openCPU extension
4. Demonstrate QlikView and R on demand.

Adding an R Script to the Open CPU Store

OpenCPU.org

As I mention earlier, OpenCPU is being developed by Jeroen Ooms. The basic philosophy of OpenCPU is to have scientific computing in the cloud. Let's now publish our first set of R code to OpenCPU by [clicking here](#).

Post a function

```
myfun <- function(foo="bar"){  
  #put a valid R function here.  
}
```

Publish Function

R Script to be published

Below is a function written in R that leverages the kmeans (klustering mean) function to cluster the data. The function takes as inputs a few parameters (e.g. *AccountCount*, *AvgOrderSize*, *CustomerCnt*) and returns back a new set of data that is clustered. Therefore, copy the code below and paste to OpenCPU link shown above. After pasting the code, press the publish function.

```
function (X, AccountCount, AvgOrderSize, CustomerCnt, Row, clusterGroup)
{
  data = data.frame(X = X, AccountCount = AccountCount, AvgOrderSize = AvgOrderSize, CustomerCnt =
CustomerCnt, Row = Row)

  cleanData <- subset(data, select = -c(X))

  fit <- kmeans(cleanData, clusterGroup)

  aggregate(cleanData, by = list(fit$cluster), FUN = mean)

  cleanData <- data.frame(cleanData, fit$cluster)

  cleanData <- subset(cleanData, select = c(Row, fit.cluster))

                                cleanData <- merge(x = data, y = cleanData, by = "Row", all.x = TRUE)

                                cleanData <- subset(cleanData, select = -c(Row))

                                return(cleanData)

}
```

Post a function

```
function (X, AccountCount, AvgOrderSize, CustomerCnt, Row, clusterGroup)
{
  data = data.frame(X = X, AccountCount = AccountCount, AvgOrderSize = AvgOrderSize, CustomerCnt = CustomerCnt, Row =
Row)
  cleanData <- subset(data, select = -c(X))
  fit <- kmeans(cleanData, clusterGroup)
  aggregate(cleanData, by = list(fit$cluster), FUN = mean)
  cleanData <- data.frame(cleanData, fit$cluster)
  cleanData <- subset(cleanData, select = c(Row, fit.cluster))
  cleanData <- merge(x = data, y = cleanData, by = "Row", all.x = TRUE)
  cleanData <- subset(cleanData, select = -c(Row))
  return(cleanData)
}
```

Publish Function

After pressing the publish button, you will be redirected to anew page as shown below:

```
{  
    "object" : "x8ed46d775e",  
    "graphs" : [],  
    "files" : {}  
}
```

Please copy down your object key (e.g. x8ed46d775e). This hashkey tells your extension where to find your function. You can think of the object as an unique street address for your function.

Once you have your object key, you have finished setting up Open CPU.

Deploy the Cluster Analysis.qvw to QlikView Server

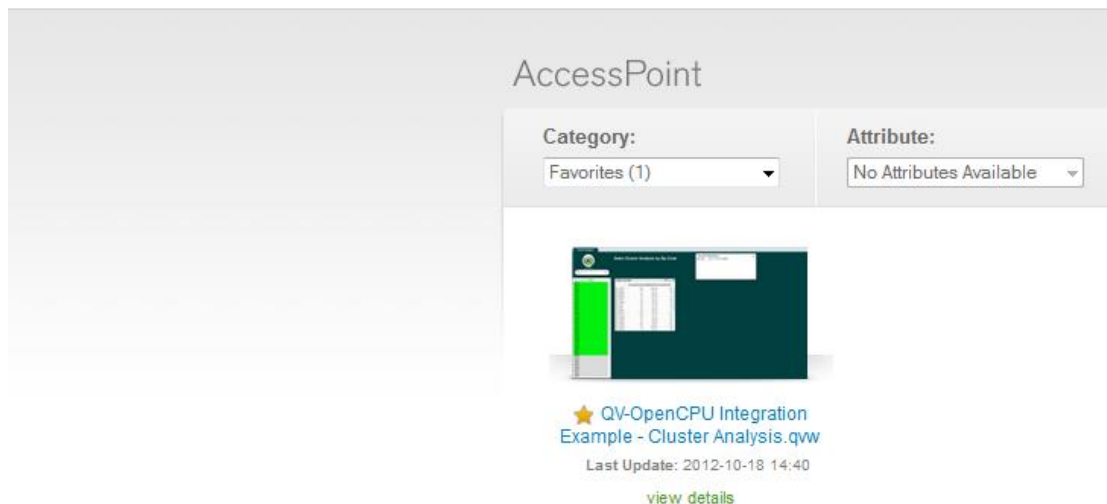
QV-OpenCPU Integration Example - Cluster Analysis.qvw

The zip file includes a QlikView document that needs to be deployed to the QlikView server. Therefore, copy and paste the application to the 'User Document'. For Windows 7, the default directory is C:\ProgramData\QlikTech\Documents. Once the file has been copied over, open up QlikView Access point using **Firefox***

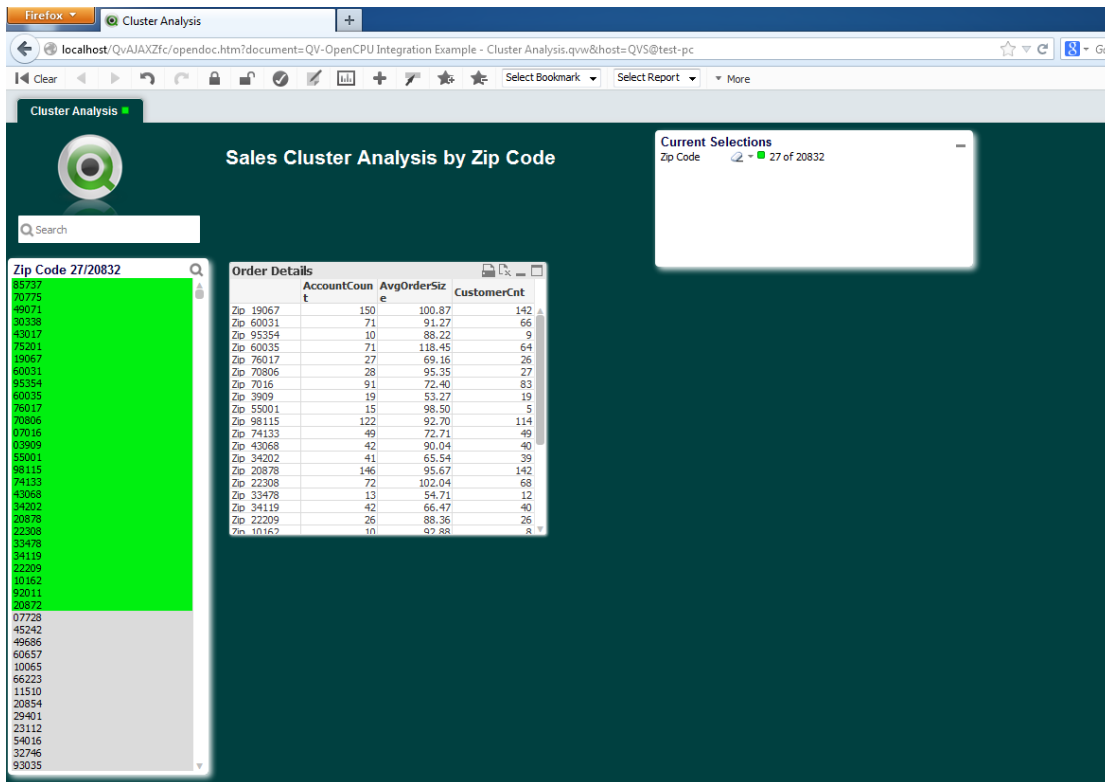
***Please note that version 1 of the extension only supports Firefox and Google Chrome.**



QlikView



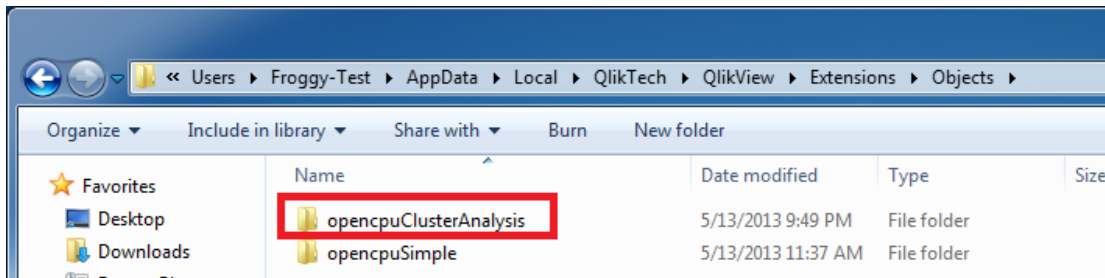
When you open the QlikView file your application should look like the following:



Install & Configure the openCPU Extension

opencpuClusterAnalysis.qar

The opencpuClusterAnalysis extension will need to be installed on the server. If you double click the opencpuClusterAnalysis.qar to install the extension, it will only install it on the client's machine. Please ensure that the extension is located at: C:\ProgramData\QlikTech\QlikView\Extensions\Objects. In this directory, there should be the opencpuClusterAnalysis folder.



Once the extension is correctly install, go to the QlikView Access point and launch the Cluster Analysis application which was describe earlier in the document. Inside the application, add a new sheet object and click on the extension objects. You will see the opencpuClusterAnalysis extensions. Drag the extension into the application.

The screenshot shows the QlikView interface for 'Cluster Analysis'. The 'New Sheet Object' menu is open, and under 'Extension Objects', 'opencpuClusterAnalysis' is highlighted. The background displays a table titled 'Cluster Analysis by Zip Code' with columns 'Zip Code', 'CustomerCnt', and 'AvgOrderSize'. A 'Current Selections' box shows 'Zip Code' with a selection of 27 of 20832.

Zip Code	CustomerCnt	AvgOrderSize
150	100.87	142
71	91.27	66
10	88.22	9
71	118.45	64
27	69.16	26
28	95.35	27
91	72.40	83
19	53.27	19
15	98.50	5
122	92.70	114
49	72.71	49
42	90.04	40
41	65.54	39
146	95.67	142
72	102.04	68
13	54.71	12
42	66.47	40
26	88.36	26
10	97.88	8

The screenshot shows the same QlikView interface, but now the 'openCPU Cluster Table' object is highlighted with a red box. The 'New Sheet Object' menu is still open, and the background table and 'Current Selections' box remain the same.

Next right click on the openCPU Cluster Table object and click on properties. The properties

dialog will open with the following options:

Dimension

This field holds the dimension of which you will cluster against

Input: **Text_ZipCode**

AccountCount

This field represents the # of distinct order number against the dimension.

Input: **count(distinct [Order Number])**

AvgOrderSize:

This field represents the average order size per dimension.

Input: **sum(Sales)/count(distinct [Order Number])**

CustomerCnt

This field represents the # of distinct customer per dimension

Input: **count(distinct Customer)**

Hashkey

This field is required to link the extension with the function hosted in openCPU.

Input: **x8ed46d775e**

ClusterGroup

This field informs the k mean cluster how to cluster the various dimensions.

Input: **4**

Properties [CH01]: openCPU Cluster Table

▼ **opencpuClusterAnalysis**

Dimension	Text_ZipCode	fx
AccountCount	count(distinct [Order Number])	fx
AvgOrderSize	sum(Sales)/count(distinct [Order Number])	fx
CustomerCnt	count(distinct Customer)	fx
hashkey	x8ed46d775e	fx
ClusterGroup	4	fx

► **Presentation**

► **Caption**

► **Options**

Once it is all set up, the extension will display a straight table.

Sales Cluster Analysis by Zip Code

Properties [CH01]: openCPU Cluster Table

opencpuClusterAnalysis

Dimension: Text_ZipCode

AccountCount: count(distinct Order Number) fX

AvgOrderSize: sum(Sales)/count(distinct Order Number) fX

CustomerCnt: count(distinct Customer) fX

hashkey: x8e4d4d775e fX

ClusterGroup: 4 fX

Current Selections

Zip Code: 27 of 20832

openCPU Cluster Table

ZipCode	AccountCount	AvgOrderSize	CustomerCnt	ClusterGroup
Zip_3909	19	53.269473684211	19	1
Zip_7016	91	72.403736263736	83	4
Zip_10162	10	92.875	8	3
Zip_19067	150	100.8746	142	2
Zip_20872	26	125.49038461538	26	3
Zip_20878	146	95.673493150685	142	2
Zip_22209	26	88.358076923077	26	1
Zip_22308	72	102.03541666667	68	4
Zip_30338	90	102.64455555556	86	4
Zip_33478	13	54.713076923077	12	1
Zip_34119	42	66.470714285714	40	1
Zip_34202	41	65.541463414634	39	1
Zip_43017	103	96.175631067961	98	4
Zip_43068	42	90.043095238095	40	1
Zip_49071	18	90.373333333333	16	3
Zip_55001	15	98.496	5	3
Zip_60031	71	91.270422535211	66	4
Zip_60035	71	118.45422535211	64	4
Zip_70775	8	125.17875	8	3
Zip_70806	28	95.35	27	3
Zip_74133	49	72.712244897959	49	1
Zip_75201	14	101.33285714286	13	3
Zip_76017	27	69.16	26	1

Demonstrate QlikView and R on demand

After the extension has been set up, click on a few zip codes and watch the straight table re-cluster each zip code on demand.

Sales Cluster Analysis by Zip Code

Order Details

Zip	AccountCount	AvgOrderSize	CustomerCnt
Zip 19067	150	100.87	142
Zip 60031	71	91.27	66
Zip 95354	10	88.22	9
Zip 60035	71	118.45	64
Zip 70017	27	69.16	26
Zip 70806	28	95.35	27
Zip 7016	91	72.40	83
Zip 3909	19	53.27	19
Zip 55001	15	98.50	5
Zip 98115	122	92.70	114
Zip 74133	49	72.71	49
Zip 43068	42	90.04	40
Zip 34202	41	65.54	39
Zip 20878	146	95.67	142
Zip 22308	72	102.04	68
Zip 33478	13	54.71	12
Zip 34119	42	66.47	40
Zip 22209	26	88.36	26
Zip 10162	10	92.88	8

openCPU Cluster Table

ZipCode	AccountCount	AvgOrderSize	CustomerCnt	ClusterGroup
Zip_3909	19	53.269473684211	19	1
Zip_7016	91	72.403736263736	83	4
Zip_10162	10	92.875	8	2
Zip_19067	150	100.8746	142	3
Zip_20872	26	125.49038461538	26	2
Zip_20878	146	95.673493150685	142	3
Zip_22209	26	88.358076923077	26	1
Zip_22308	72	102.03541666667	68	4
Zip_30338	90	102.64455555556	86	4
Zip_33478	13	54.713076923077	12	1
Zip_34119	42	66.470714285714	40	1
Zip_34202	41	65.541463414634	39	1
Zip_43017	103	96.175631067961	98	4
Zip_43068	42	90.043095238095	40	1
Zip_49071	18	90.373333333333	16	2
Zip_55001	15	98.496	5	2
Zip_60031	71	91.270422535211	66	4
Zip_60035	71	118.45422535211	64	4
Zip_70775	8	125.17875	8	2
Zip_70806	28	95.35	27	2
Zip_74133	49	72.712244897959	49	1
Zip_75201	14	101.33285714286	13	2
Zip_76017	27	69.16	26	1

With 27 zip codes selected

Sales Cluster Analysis by Zip Code

Current Selections
Zip Code 14 of 20832

Zip Code 14/20832

Zip	AccountCount	AvgOrderSize	CustomerCnt
Zip_19067	150	100.87	142
Zip_60031	71	91.27	66
Zip_95354	10	88.22	9
Zip_60035	71	118.45	64
Zip_76017	27	69.16	26
Zip_70806	28	95.35	27
Zip_7016	91	72.40	83
Zip_3909	19	53.27	19
Zip_30338	90	102.64	86
Zip_43017	18	90.37	16
Zip_85737	29	66.11	29
Zip_43017	103	96.18	98
Zip_70775	8	125.18	8
Zip_75201	14	101.33	13

ZipCode	AccountCount	AvgOrderSize	CustomerCnt	ClusterGroup
Zip_3909	19	53.269473684211	19	3
Zip_7016	91	72.403738263736	83	1
Zip_19067	150	100.8746	142	2
Zip_30338	90	102.644555555556	86	1
Zip_43017	103	96.175631067961	98	1
Zip_49071	18	90.3733333333333	16	4
Zip_60031	71	91.270422535211	66	1
Zip_60035	71	118.45422535211	64	1
Zip_70775	8	125.17875	8	4
Zip_70806	28	95.35	27	4
Zip_75201	14	101.33285714286	13	4
Zip_76017	27	69.16	26	3
Zip_85737	29	66.106896551724	29	3
Zip_95354	10	88.219	9	4

With 14 Zip codes selected

Appendix

Below is the process workflow on the exchange between R, OpenCPU and the extension.

