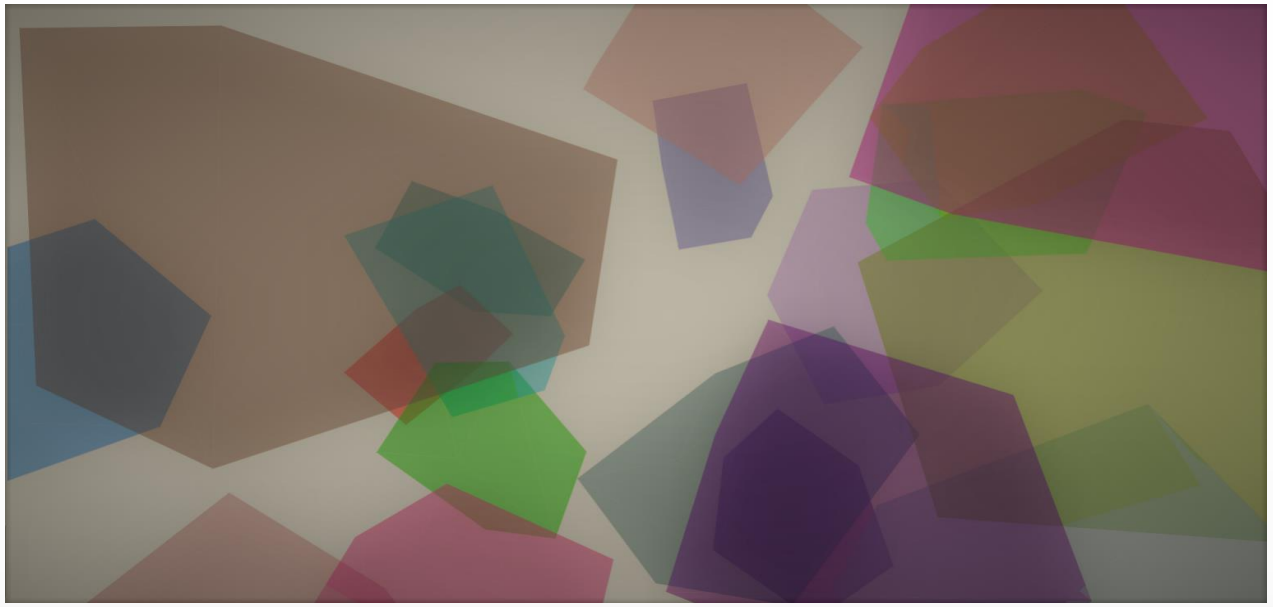
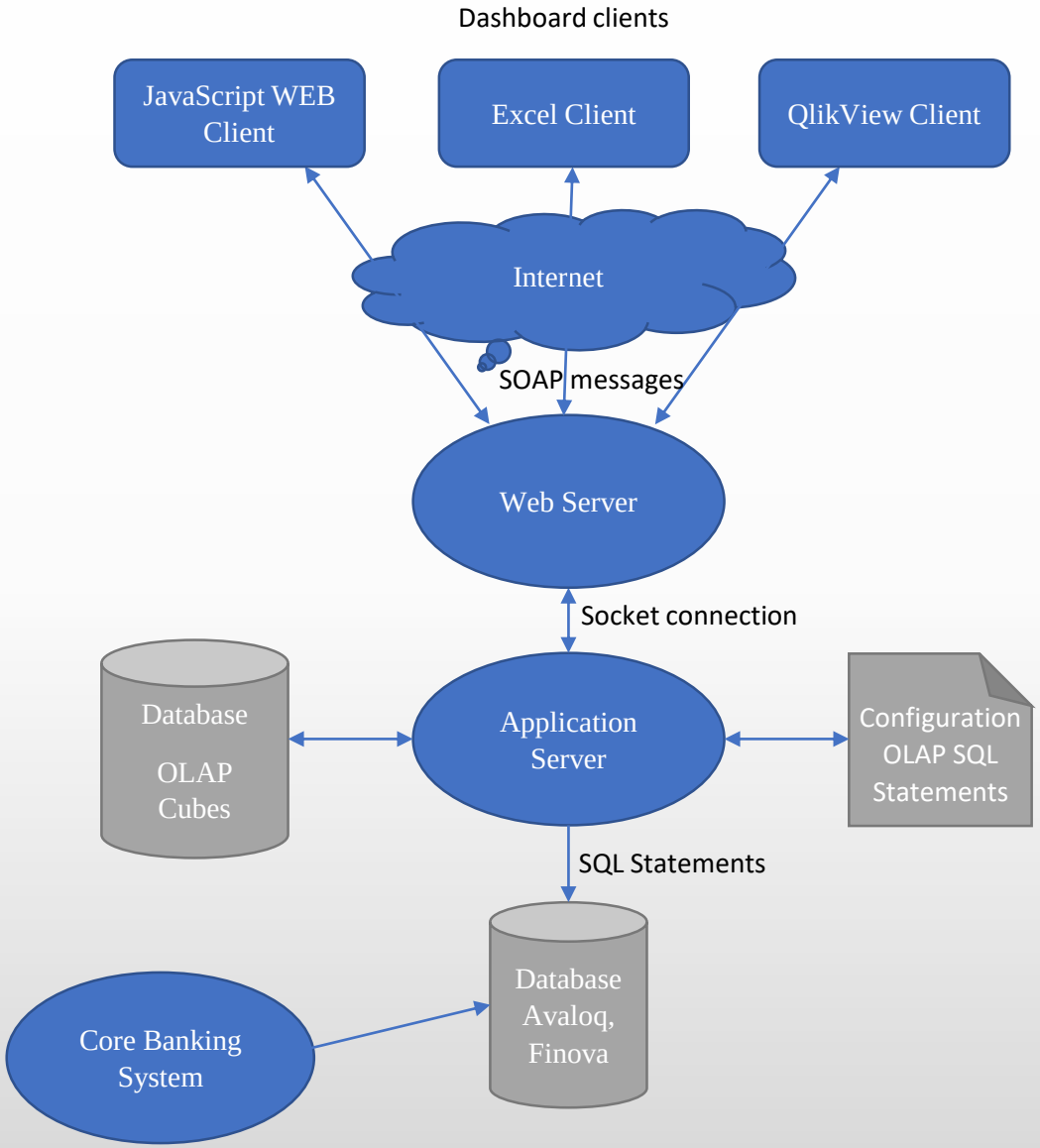


Visualization of Financial Data

Dr. Anatoliy Antonov, Ivan Bogdanov
Eurorisk Systems Ltd.

Eurorisk Systems Ltd
General Kiselov Str. 31
9002 Varna, Bulgaria
Tel. +359 52 612 367
Fax +359 52 612 371
E-mail: info@eurorisksystems.com
Web: www.eurorisksystems.com

Application Architecture Diagram

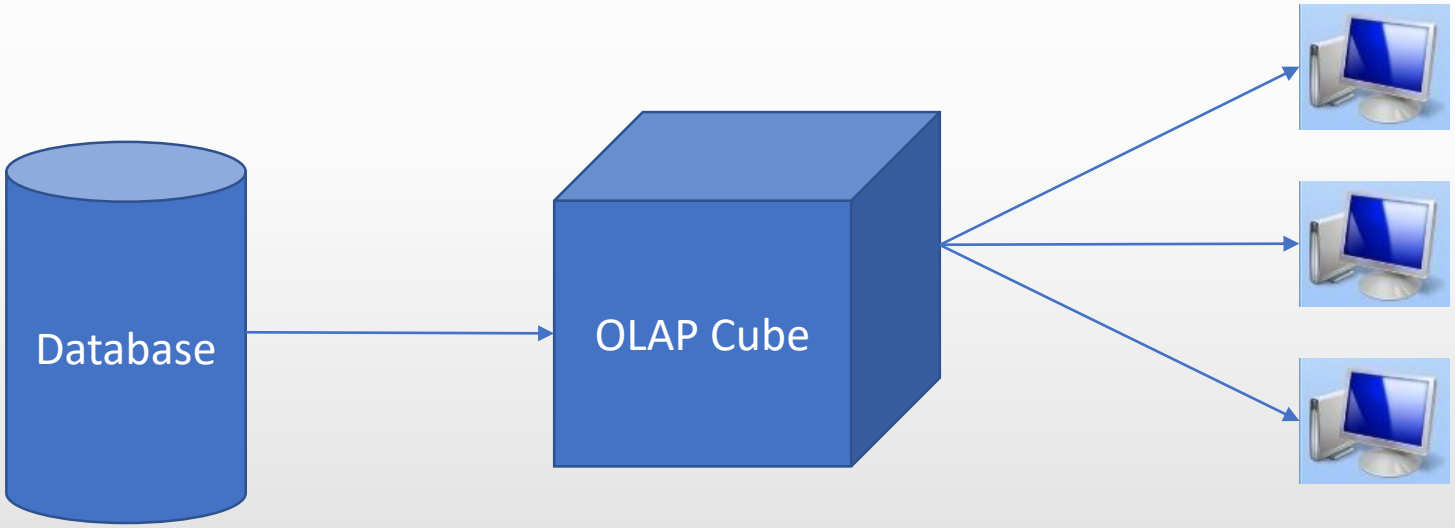


To visualize financial data generated from existing Core Banking System one needs the components shown in the diagram left. It is considered that the Core Banking System and its database already exists so following components are needed:

- Application Server to extract data from Core Banking Database and to generate Visualization OLAP Cubes
  - SQL Configuration for the OLAP Cube Load Scripts,

    ETL (Extract Transform Load) tools can be used too
  - Database to store the created visualization information
- Web Server to exchange commands and data between the clients and the Application Server via SOAP and Sockets
- Available WEB Clients which can represent Dashboards:
  - In JavaScript or in Excel using Pivot and WEB Services
  - In QlikView using the Visualization and WEB Services

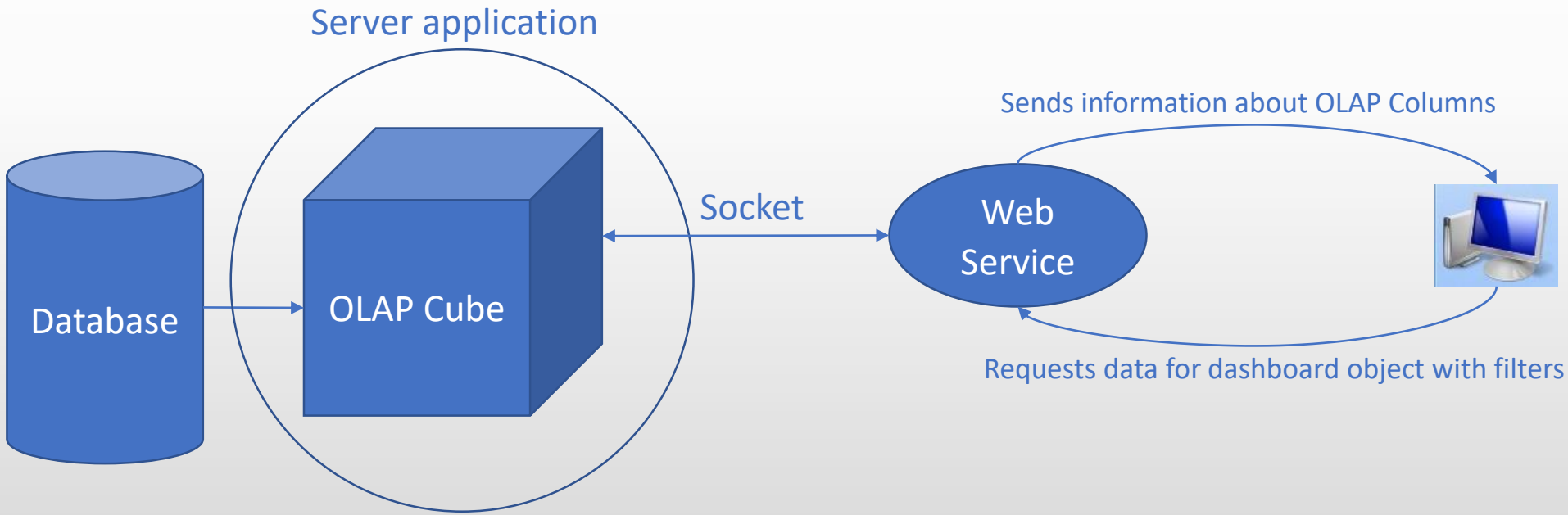
The main porpoise of the application server is to access the raw data from the database, generate visualization information as OLAP Cubes and send Sub-Cubes according to commands to the Web server via XML-defined socket protocol. The application server also needs a configuration for the OLAP Cube Load Scripts in which is stored information for the database structure and also SQL statements for generating the OLAP Cube. This is a typical ETL (Extract Transform Load) procedure, so ETL Toll are applicable too, see last slide



The Web Server is communicating with multiple client instances by Internet via SOAP/WSDL, i.e. via WEB Services. It receives commands from the specific client application, requests data from the application server (which have access to already prepared OLAP Cubes) and returns a response message including visualization data to the client. Some of the current functionalities of the application server are:

- getOLAPCubes – returns all available OLAP Cubes as a list
- getCubeMetaData – returns all available columns from the Cube and its types
- getCubeDataByColumnsAndFiltered – returns to the Dashboard specific visualization data (Sub-Cubes) from existing Cube, filtered by column and row selection
- import/export/delete Cubes of the application server database
- manage the OLAP Cube SQL Load Scripts

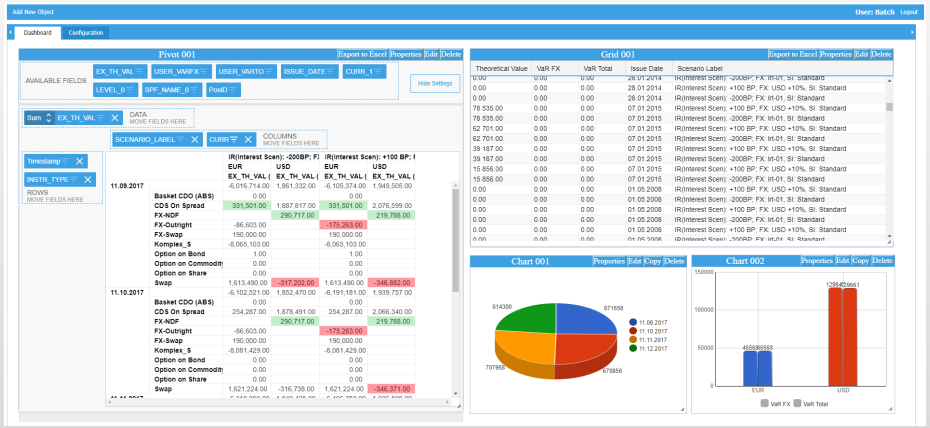
Example workflow of the Web Service and connection to application server and clients is given below:
At first information about all OLAP Columns available is sent via Web Service to the Client. User selects the columns and rows and filters that want to apply on the dashboard and sends request for building the corresponding dashboard object. The Web server processes the request to the application server which apply the filters over the OLAP Cube and returns the data to the Web Server which transfers it to the client application.



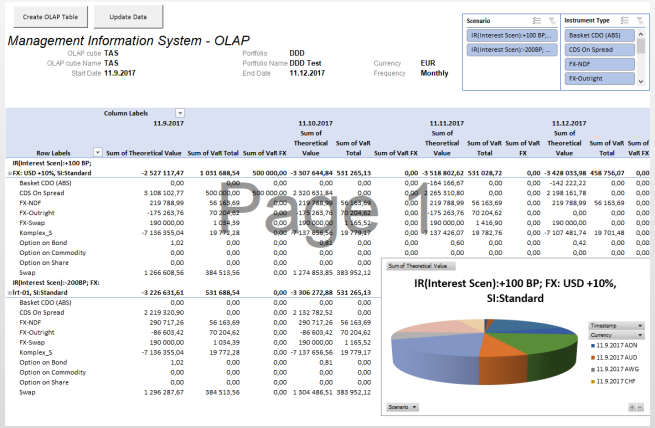
The Client application solutions can be made in different ways. Some proposals are:

- JavaScript WEB Client – JavaScript based GUI developed with DHTMLX library, deployed on a WEB Server and can be access via browser. 3D interactive graphic can be added in addition. Both tools are of low cost or cost free
- Excel Document Client – Excel Document including Pivot Tables which is designed to work as a client of the Web Server and which runs according to same protocol as the JavaScript WEB Client
- QlikView Client – It is also possible to provide a client using the visualization features and WEB Service scripts of the QlikView tool in non-server mode

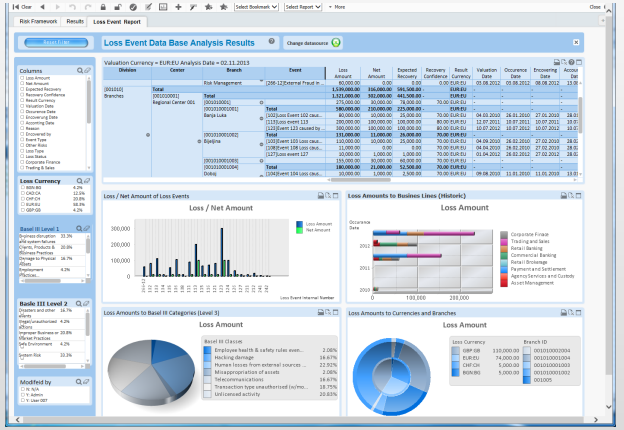
JavaScript WEB client Dashboard



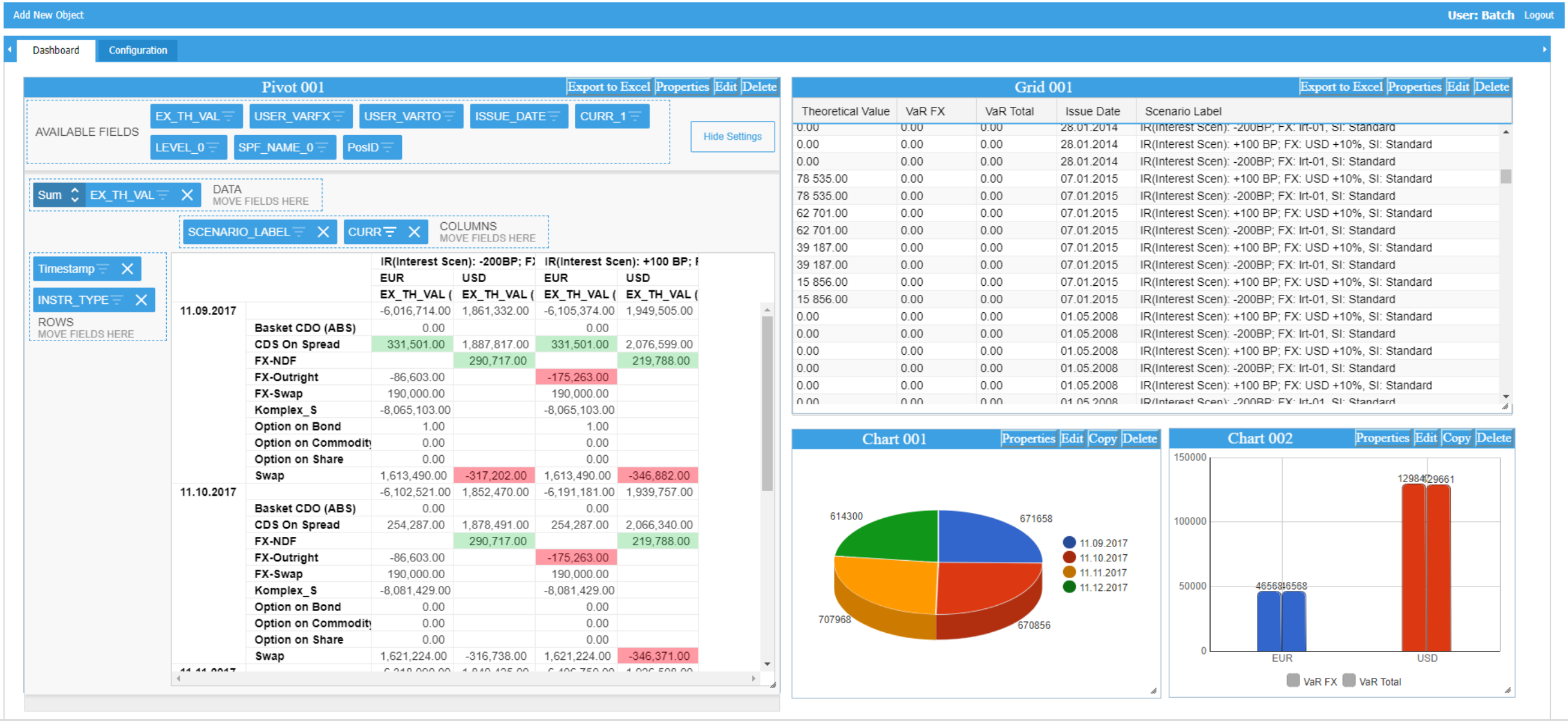
Excel WEB client incl. Pivot Dashboard



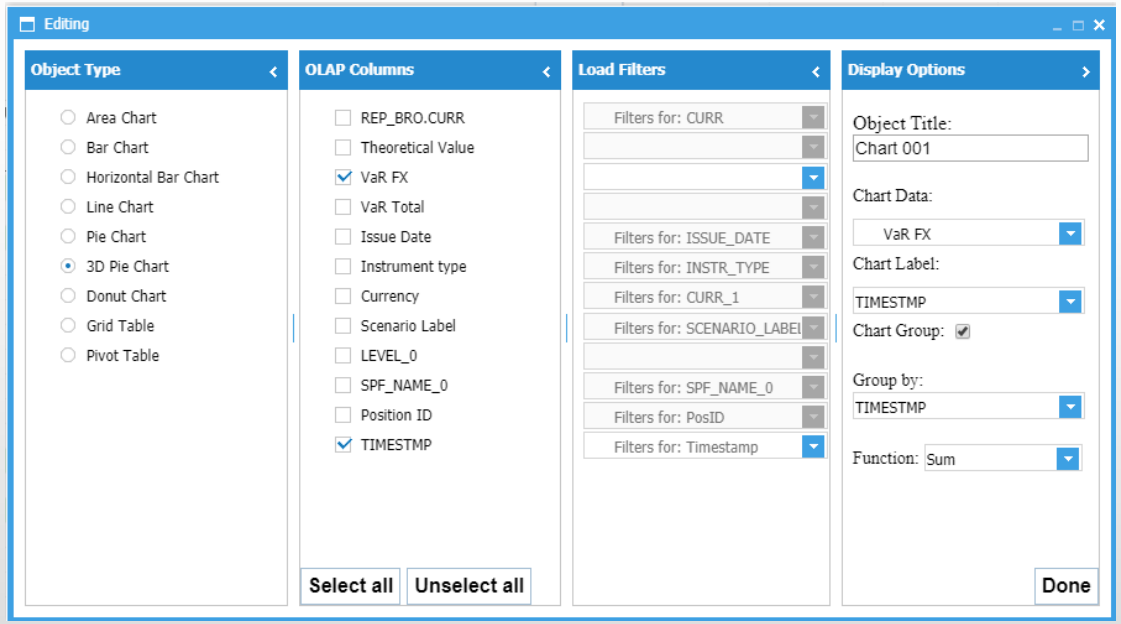
QlikView WEB client Dashboard

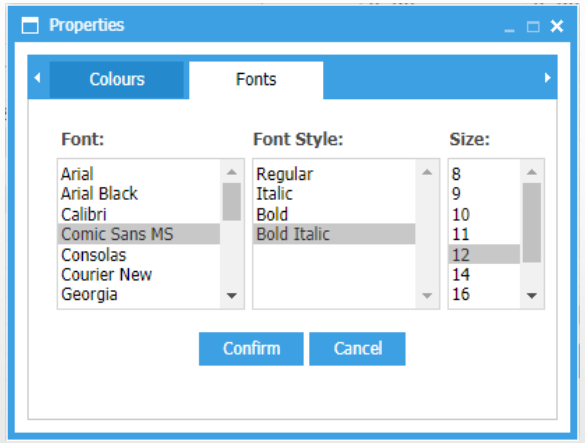


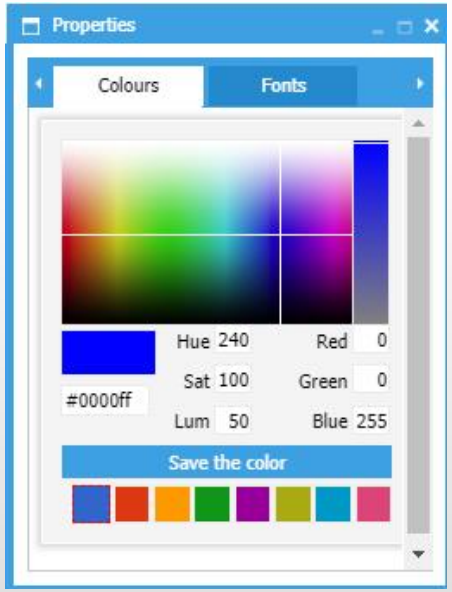
JavaScript WEB client Dashboard – OLAP, tables, pivots, 2D - charts



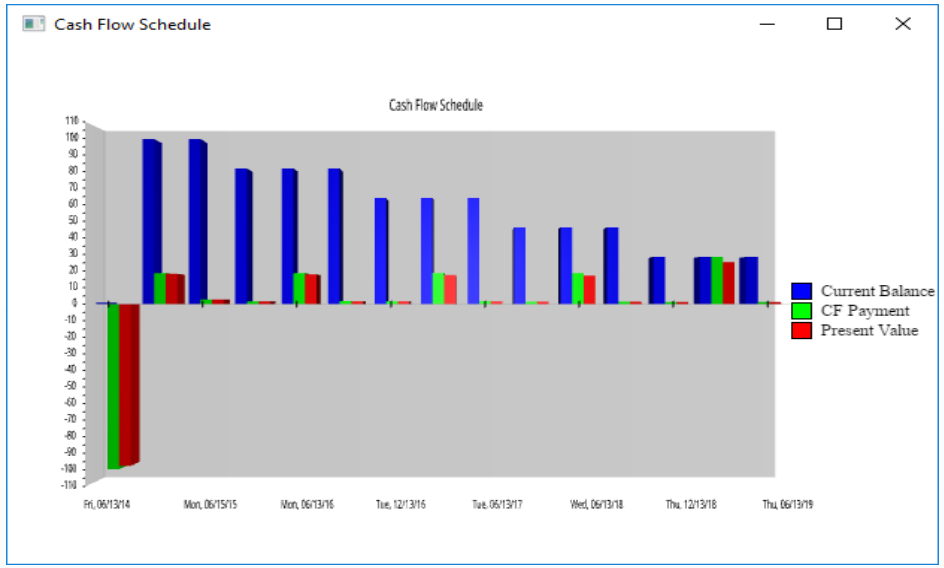
The WEB graphical user interface is JavaScript based and developed with DHTMLX library. It can be accessed via any WEB browser, tablet or smartphone. It has a Configuration tab where is specified the settings and filtering of the OLAP Cube and a Dashboard tab where is visualized the data. You can dynamically add, edit or remove dashboard elements with request for data from the server. Each element has its own settings such as OLAP columns, Load row filters, chart colors, font style, font size. You can customize your dashboard and store its settings to external database so you can load it there after.

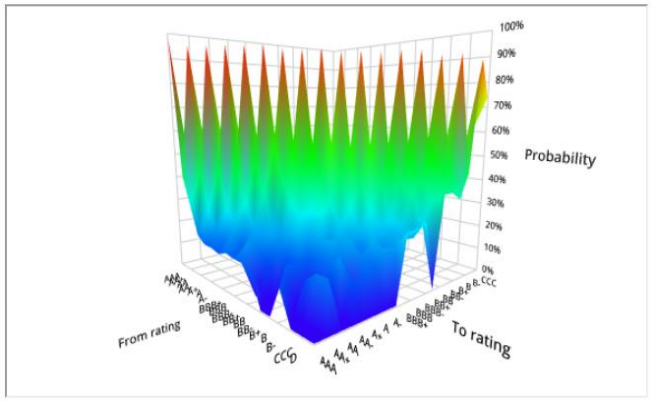


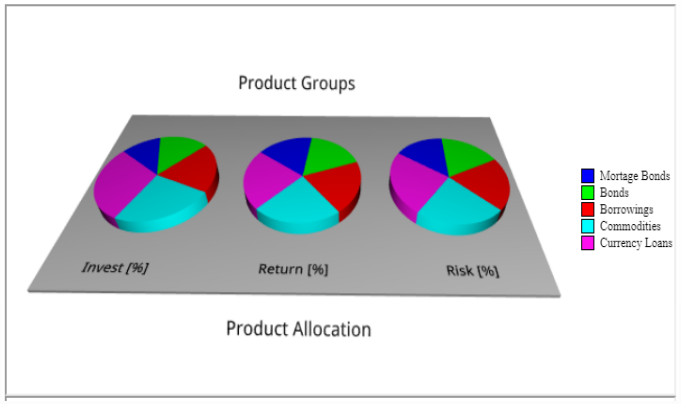


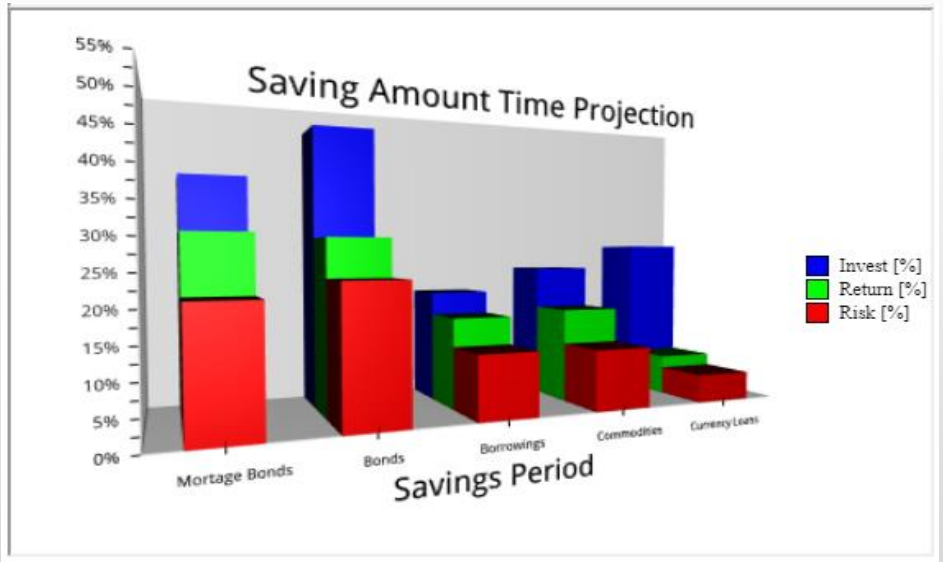


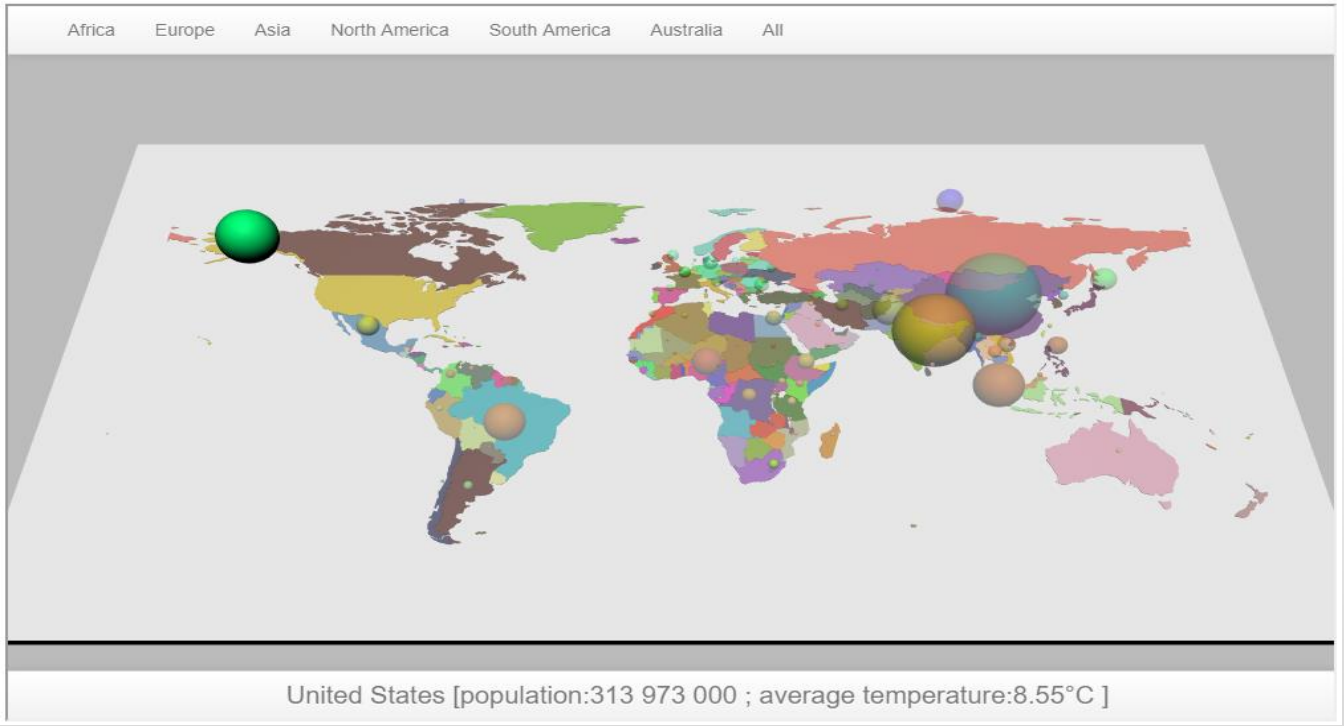
JavaScript WEB client – additional 3D interactive charts: zoom, move, rotate











Excel client including Pivot Tables

Create OLAP Table | Update Data

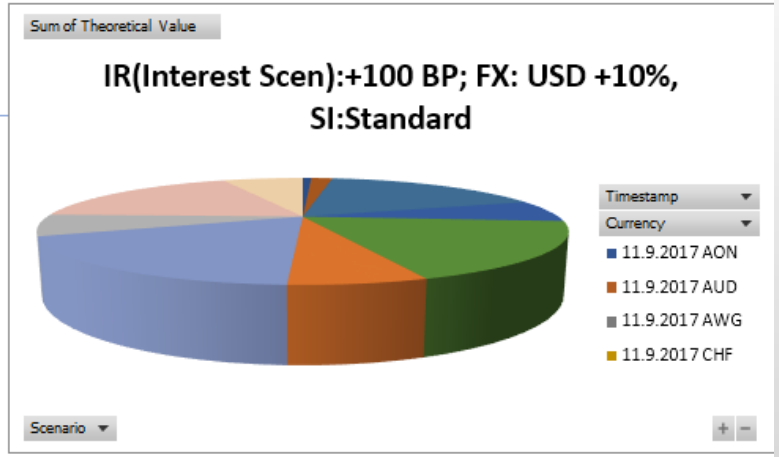
Management Information System - OLAP

OLAP cube	TAS	Portfolio	DDD		
OLAP cube Name	TAS	Portfolio Name	DDD Test	Currency	EUR
Start Date	11.9.2017	End Date	11.12.2017	Frequency	Monthly

Scenario: IR(Interest Scen):+100 BP;... | IR(Interest Scen):-200BP; ...

Instrument Type: Basket CDO (ABS) | CDS On Spread | FX-NDF | FX-Outright

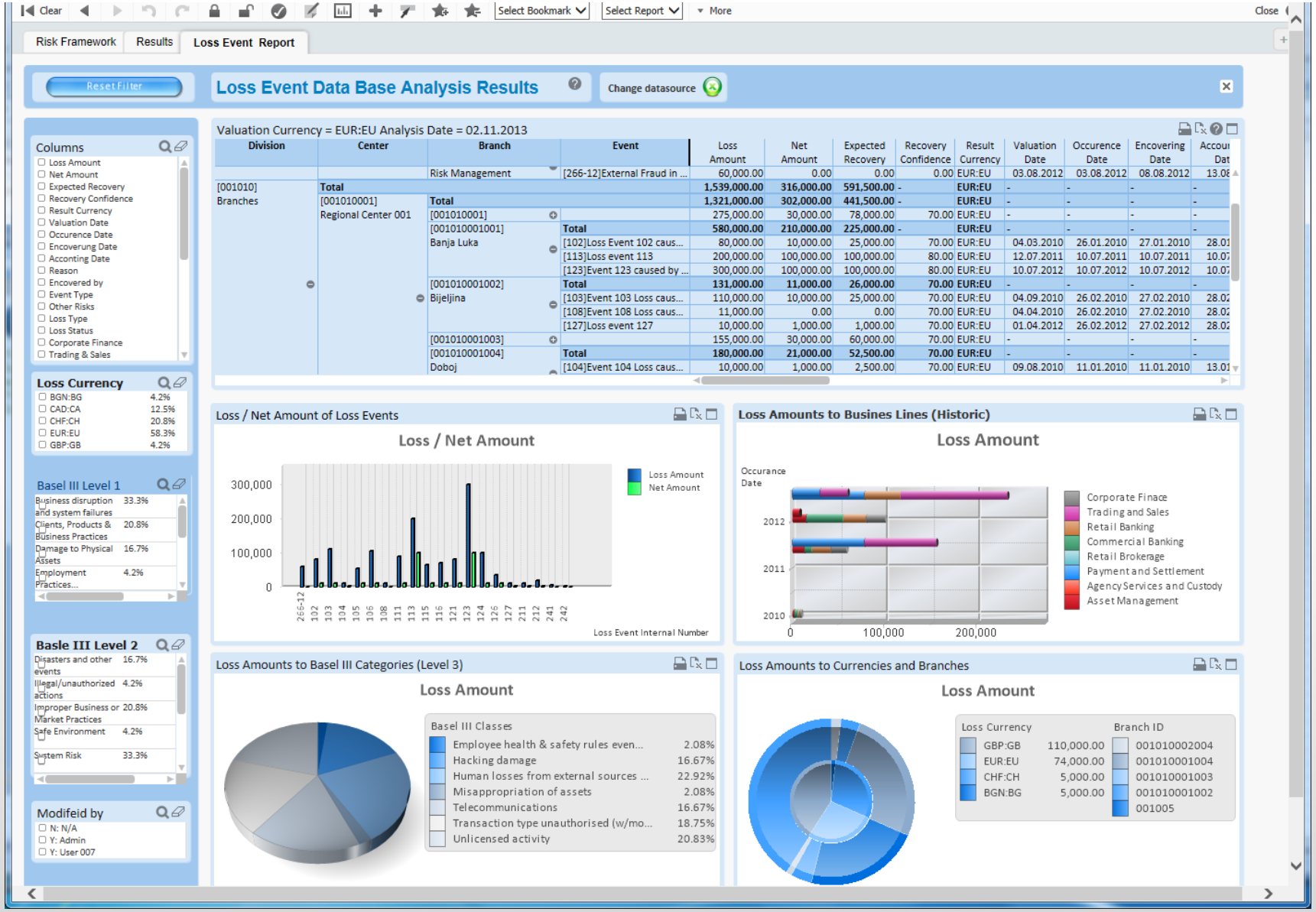
Column Labels	11.9.2017			11.10.2017			11.11.2017			11.12.2017		
Row Labels	Sum of Theoretical Value	Sum of VaR Total	Sum of VaR FX	Sum of Theoretical Value	Sum of VaR Total	Sum of VaR FX	Sum of Theoretical Value	Sum of VaR Total	Sum of VaR FX	Sum of Theoretical Value	Sum of VaR Total	Sum of VaR FX
IR(Interest Scen):+100 BP; FX: USD +10%, SI:Standard	**-2 527 117,47**	**1 031 688,54**	**500 000,00**	**-3 307 644,84**	**531 265,13**	**0,00**	**-3 518 802,62**	**531 028,72**	**0,00**	**-3 428 033,98**	**458 756,07**	**0,00**
Basket CDO (ABS)	0,00	0,00	0,00	0,00	0,00	0,00	-164 166,67	0,00	0,00	-142 222,22	0,00	0,00
CDS On Spread	3 108 102,77	500 000,00	500 000,00	2 320 631,84	0,00	0,00	2 265 310,80	0,00	0,00	2 198 161,78	0,00	0,00
FX-NDF	219 788,99	56 163,69	0,00	219 788,99	56 163,69	0,00	219 788,99	56 163,69	0,00	219 788,99	56 163,69	0,00
FX-Outright	-175 263,76	70 204,62	0,00	-175 263,76	70 204,62	0,00	-175 263,76	70 204,62	0,00	0,00	0,00	0,00
FX-Swap	190 000,00	1 034,39	0,00	190 000,00	1 165,52	0,00	190 000,00	1 416,90	0,00	190 000,00	0,00	0,00
Komplex_S	-7 136 355,04	19 772,28	0,00	-7 137 656,56	19 779,17	0,00	-7 137 426,07	19 782,76	0,00	-7 107 481,74	19 701,48	0,00
Option on Bond	1,02	0,00	0,00	0,81	0,00	0,00	0,60	0,00	0,00	0,42	0,00	0,00
Option on Commodity	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
Option on Share	0,00	0,00	0,00	0,00	0,00							
Swap	1 266 608,56	384 513,56	0,00	1 274 853,85	383 952,12							
IR(Interest Scen):-200BP; FX: Irt-01, SI:Standard	**-3 226 631,61**	**531 688,54**	**0,00**	**-3 306 272,88**	**531 265,13**							
Basket CDO (ABS)	0,00	0,00	0,00	0,00	0,00							
CDS On Spread	2 219 320,90	0,00	0,00	2 132 782,52	0,00							
FX-NDF	290 717,26	56 163,69	0,00	290 717,26	56 163,69							
FX-Outright	-86 603,42	70 204,62	0,00	-86 603,42	70 204,62							
FX-Swap	190 000,00	1 034,39	0,00	190 000,00	1 165,52							
Komplex_S	-7 136 355,04	19 772,28	0,00	-7 137 656,56	19 779,17							
Option on Bond	1,02	0,00	0,00	0,81	0,00							
Option on Commodity	0,00	0,00	0,00	0,00	0,00							
Option on Share	0,00	0,00	0,00	0,00	0,00							
Swap	1 296 287,67	384 513,56	0,00	1 304 486,51	383 952,12							



Features of the Excel client pivot table:

- Work with dimensions and aggregable columns
- Filters for dimensions
- Aggregation of calculated results
- Drop-Down-Feature
- Excel Charts
- Macros in Visual Basic Script
- WEB Services support

QlikView client - visualization features



Features of the QlikView client:

- Work with dimensions and aggregable columns
- Filters of dimensions
- Aggregation of calculated results
- Drop-Down-Feature
- Various charts and table representations
- Selection of table columns
- Macros in JavaScript
- WEB Services support

To become fully working systems there are few steps of installation:

1. Place the application server on server machine in a LAN with access to the database of the Core Banking System
2. Setup the configuration with information for connection to the database of the Core Banking System and create scripts to extract visualization data into OLAP cubes
3. Install a WEB Server (we suggest Apache Tomcat)
4. Deploy the Web Service package on the WEB Server
5. Configure the settings of the Web Service package
6. You receive there after access to the application via the WEB clients:
   - WEB application via browser and WEB Services or
   - Desktop application in Excel or in QlikView which connects using WEB Services

One of the main task of the application server is to prepare the OLAP cubes via access to the database tables of the Core Banking System using SQL load scripts which extract the visualization data from database of the Core Banking System, transform and reorganize it and load it as OLAP cubes. This is a typical task for configurable ETL (Extract Visualize Load) tools, so this task can be performed alternatively using ETL. An example transformation and a list of provider of ETL integration tools are given below, many of the tools are at low cost or cost free.