In-memory databases and innovations in Business Intelligence

Ruxandra BĂBEANU, Marian CIOBANU
University of Economic Studies, Bucharest, Romania
babeanu.ruxandra@gmail.com, marianciobanu6@yahoo.com

The large amount of data that companies are dealing with, day by day, is a big challenge for the traditional BI systems and databases. A significant part of this data is usually wasted because the companies do not own the appropriate capacity to process it. In the actual competitive environment, this lost data could point up valuable information if it was analyzed and put in the right context. In these circumstances, in-memory databases seem to be the solution. This innovative technology combined with specialized BI solutions offers high performance and satisfaction to users and comes up with new data modeling and processing options.
***Keywords**: Business Intelligence, in-memory database, SAP HANA, data modeling, text processing*

1 Introduction

Business Intelligence solutions are not a novelty any more for the big companies and not only for these ones. The competitive advantages given by the usage of BI tools are significant. Having at the right time a report concerning a certain region or a certain product sales, or a report predicting the way a new product launch on the market will influence the company, brings a great advantage against competition. This will materialize quickly in revenues and in a higher trust level of the consumer.

Lots of companies, even though they have implemented Data Warehouses and Business Intelligence solutions, have big problems in obtaining the data in time. There are cases where the execution of a report takes between a few hours to several days, and the time lost for this execution influences, directly or indirectly, the profit. Therefore, taking the right decision at the right time, meaning before the competitors, definitely represents an important move on the market.

The hard work in obtaining the data at the right moment is, in most of the cases, due to a combination of hardware support and inappropriate software. Standard databases store the data on disc and the I/O operations are very slow compared to those made in RAM memory.

The need of processing large amounts of data very fast was the main reason that led to the development of in-memory databases. These databases revolutionized the entire Business Intelligence area and managed to bring outstanding results exactly where the classic databases failed.

2. General characteristics of in-memory databases

As their name suggest, the in-memory databases stop storing the data at disk level, moving instead the storage to the memory. This change comes with some great advantages, but also brings a set of risks.

Considering that the main characteristic of these DBMSs is the memory storage, which can lead to amazing processing speeds, there is also a big risk to manage, and that is the risk in loosing data in case of an unplanned restart of the system or in case of problems with the power. Therefore, for these type of databases the ACID (Atomicity, Consistency, Isolation, Durability) concept is not fully accomplished [1]. The problem appears at the last property, *durability*, which is not fully satisfied. There are several solutions that can assure the *persistency of data*, some of them being already implemented by the main in-memory databases producers [1].

From a *hardware perspective*, the following solutions can be implemented:

- use of NVDIMM (non-volatile dual in-line memory module) memory, a special type of memory that has the property of keeping the data in case

of an unexpected incident, such as an unexpected system restart or an electric power failure;
- use of NVRAM (non-volatile random access memory) memory, which is a special type of memory having similar properties as NVDIMM memory type.

From a *software perspective*, there are also several solutions that can be implemented to assure the durability of data:
- use of *a journal file*, which store a log of all transactions in the database;
- use of "*high availability*" implementation, which suppose replication of the database on a secondary database, and, in case of major issues of the primary database, the standby database will replace automatically the primary database; it is recommended to use this method along with any other method mentioned above, in order to assure complete recovery in case of system crash;
- use of *snapshot files*; these files keep the state of the database at different moments of time, but the major disadvantage in this case is that the most recent changes will not be captured. The recommended approach in this case is to use this method concurrent with any other method mentioned above.

The advantages brought by an in-memory database are, first of all, due to the fact that the slow I/O disk operations don't exist anymore. Here a few outlines regarding the in-memory databases:
- very *fast processing* of large amounts of data;
- *reduced time of blocking the records*, because all the insert, update and delete operations are made in real time;
- *fast data queries*.

Regarding the ***data access method***, there are defined different indexing methods, based either on hash functions or on several tree types. *T-tree* is a special tree type, created particularly for these kind of databases. It is a balanced tree and it is especially optimized for indexing the data stored exclusively in memory. The main characteristic of this tree is that the *index is not storing any effective data*, is storing only a pointer to the data, because all is kept in memory.

In-memory databases use for ***data representation*** the column store model. This leads to an efficient data store, because for building the relational tuples they use the pointers. This way the effective value is stored only once in a dictionary and, instead of storing the same value many times in the memory, the tuples will be built by storing only the pointer to that value. If we have to deal with large values appearing many times in the database, the storage space will be this way significantly improved.

The ***performance*** is the key characteristic for the in-memory databases. This is not any more influenced by the slow I/O operations from the memory to the disk and backwards, it is just based on the processing speed. Therefore, an appropriate hardware will certainly assure an optimum response time of the database.

In order to assure the ***physical integrity*** of data, the *backup files* must be saved on a safe storage support. In case of a major incident, the database has to be recovered from the last backup file and afterwards has to be updated using the *journal file*, which logs all the transactions in the database.

3. Short history

The need of performance in the IT domain combined with the advantages of in-memory computing are the main factors that influenced the appearance of in-memory databases. An in-memory DBMS uses the memory as the main storage support, compared to the classic DBMSs that use the disk as the main storage place.

The first in-memory databases appeared in the early '90s. In 1992, White Cross Systems Limited Company offered the first version of this type of DBMS, sold nowadays as "Kognitio Analytical

Platform", which was firstly delivered on proprietary hardware. In1993, the company Perihelion Software launched the in-memory DBMS called "Polyhedra" [1],[3].At that time, Polyhedra was designed to "keep the working copy of the data in memory" and it was based on a client- server architecture [1]. Polyhedrais now sold by ENEA AB.

In 2001 was released eXtremeDB, belonging to McObject company. It was an advanced DBMS, conceived for in-memory storage and designed to use minimal CPU and memory resources. It was designed especially for the real-time embedded systems [1].

The main in-memory database producers nowadays are:

- Oracle, with Oracle Database 12c – In Memory option, released in 2014;
- SAP, with SAP HANA (High - Performance Analytical Appliance), launched in 2010;
- IBM, with DB2 BLU, released in 2013;
- Microsoft, with SQL Server 2012 (actual SQL Server 2014), released in 2012.

4. Main characteristics of SAP HANA

SAP HANA is a complex technology, developed in C++ and running on SUSE Linux Enterprise Server, capable of fast processing of large amounts of data in real time. Also, it is a *platform* used to support the developing of real time applications and, not least, it is an in-memory database, which can sustain a large data warehouse.

SAP HANA, as an in-memory database, comes with some special features, using an innovative hardware and software. It offers the possibility of analysing big volumes of data and the flexibility in analysing various types of data, not only the traditional ones.

SAP HANA must be seen as a complex platform, designed to sustain and improve all the business processes. It can handle not only the Analytics and Business Intelligence processes, but also the OLTP transactions, as it can be delivered as support for SAP ERP.

Regarding the architecture, in *Figure.1 – HANA Architecture*, we can observe the main components of SAP HANA platform [5].

HANA is a system that has the in-memory database as the foundation. To this, it can be added several components and add-ons to support various functionalities: real-time data replication, monitoring instruments, release management instruments etc.

HANA database includes four dedicated servers, as follows [5], [7]:

- *IndexServer* – the main component because it is the place where the data is effectively stored; it also contains the data processing engines;
- *Preprocessor* – the server used in case of text processing and text analysis;
- *NameServer* – holds the information about the system landscape and in a distributed system its main responsibility is to locate the data;
- *StatisticsServer* – keeps information about the performance parameters: system status, CPU usage, memory usage etc.

To be noticed that, in case of a multi-node cluster configuration, the four servers can be found on each node.

The main capabilities HANA offers are stated below:

- *optimized analytics data models*: Analytic View, Attribute View, Calculation View etc.
- *removal of data caching*;
- taking over the *complex operations pushed at database level* by the specially optimized BI client tools which have as source data HANA database;
- including the *SAP HANA predictive* functionality.

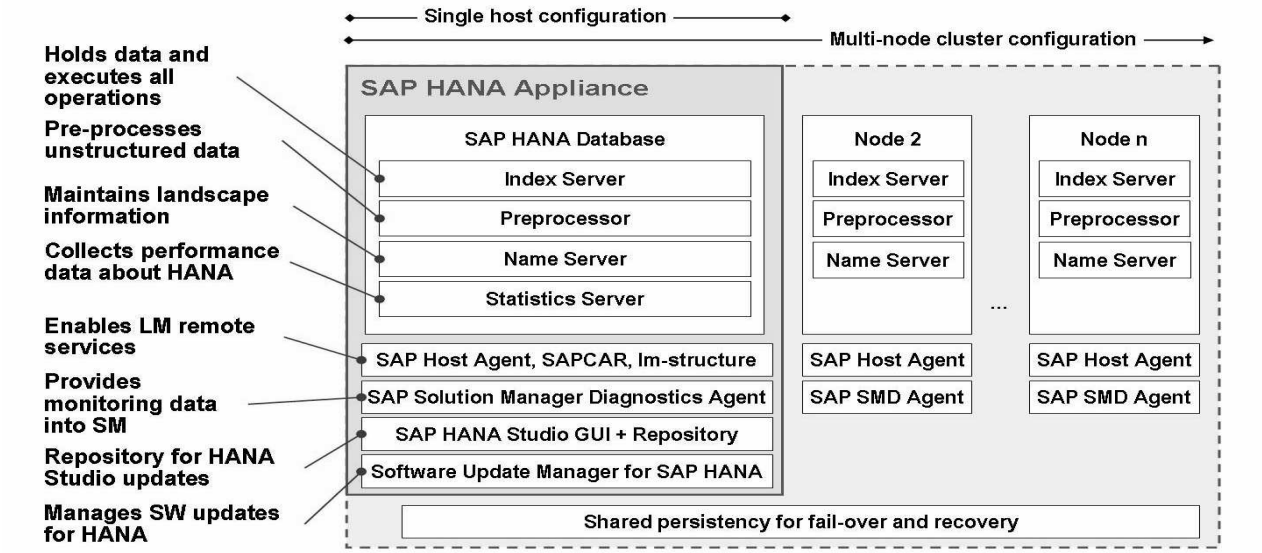

Fig. 1. Hana Architecture

5. Business Intelligence with Sap HANA

Considering the capabilities HANA possesses, the performance of the BI solutions which have as source data this database are remarkable. SAP created some special functionalities for the BI solutions, some of them specially optimized for SAP HANA.

The *interface* through which the user interacts with the system is SAP HANA Studio. This working environment is based on Eclipse and offers the user the possibility of working in several perspectives (Information modeling, Application development, Administration, Monitoring and security etc.).

Before to get to the data modeling step, first we have to bring the data in HANA. An overview regarding the data acquisition in HANA can be found in *Figure 2 – Different ways for data acquisition in HANA* [5].

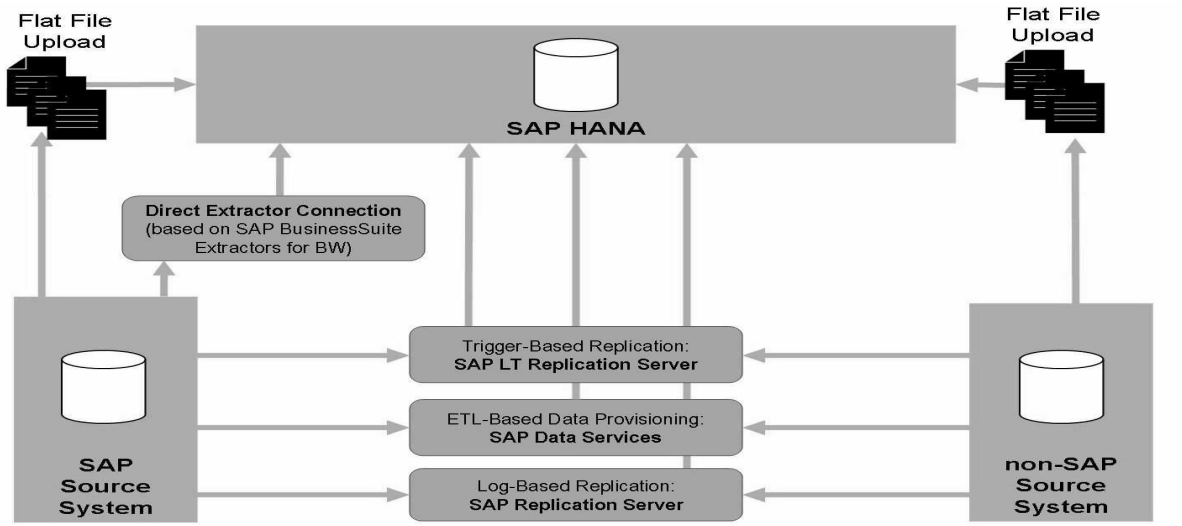

Fig. 2. Different ways for data acquisition in Hana

Loading the data into HANA can be done in one of the following options:

- Direct load from *flat files* (*xls*,*.xlsx* or *.csv)* through an interface accessible from HANA Studio;
- *SAP Data Services*–software solution for data integration from various sources, for data quality management and for text analysis;

- *SAP Landscape Transformation* – tool used for real-time replication of data, either from SAP source system or from non-SAP source system;
- *SAP Direct Extractor connection* –tool used to import data from SAP Business Suite using specific connectors.
- *SAP Replication Server* – advanced tool used for transferring and synchronizing the data across the organization.

The data can be stored in "column-based" tables or in "row-based" tables; SAP's recommendation is to use the tables in the first category in order to assure high level of performance.

For the data modeling, in HANA Studio the Modeller Perspective must be accessed. The modeling objects available in HANA are:

- ***Attribute Views***– reusable objects, used for *describing the data*; attribute views "*are used to give context. This context is provided by text tables, which give meaning to data*" [6]. They can contain dimensions, hierarchies and calculated attributes. For example, in the banking domain, if we have a fact table containing the IDs of the customers, we could use an attribute view to make available more information about the customer, such as name, age, monthly revenue etc. In this way, the IDs acquire a meaning and are framed within a context;
- ***Analytic Views*** – they offer the possibility of data modelling using measures (numeric values) and aggregations. These views can contain *Attribute views*, which are linked to the fact table according to the *star schema mode.* In SAP HANA, these are the fastest modeling objects, but the limitation is that an analytic view can be created using only one fact table [6];
- ***Calculation Views*** – objects that offer the greatest flexibility in data modeling. These are similar to Analytic views and offer the possibility to define complex calculation. The main difference as against analytic views is that the calculation views offer the option to *join various fact tables* for using different measures. Furthermore, there are two ways of creating this type of objects: *graphical modeling* or *scripting modeling* in case there are complex operations that cannot be covered by the graphical model.
- ***Decision Tables*** – objects that allow business rules modeling regarding decisions, in a certain context. These objects use the decision trees logics. A decision table can be created using an existing table in HANA, which will represent in this case the data source of the object. Next step is to define the attributes that will be considered in the decision, along with their associated conditions and the action that must be taken when the defined conditions are fulfilled.

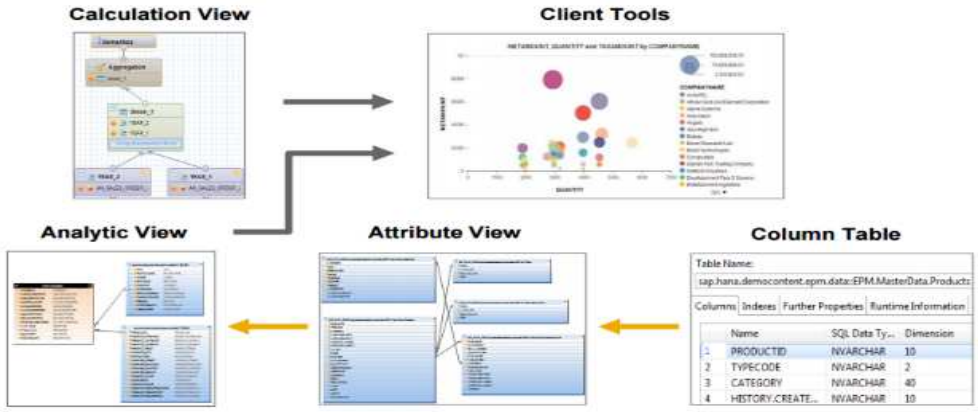

Fig. 3. Query views in Hana

Figure 3- Query views in HANA offers a better view of how all of these analytic objects can be used [6].

If the user' requirements are not fulfilled by the graphic modeling options, HANA offers the possibility to create stored procedures to define the desired logic, using SQLScript languange.

SAP HANA has 3 different engines used for processing the queries, each one used depending on the queried object, as can be seen in *Figure 4- Processing engines in HANA* [8]:

- **Join engine** – used when an Attribute View is queried;
- **OLAP engine** – used when a query based on an Analytical View is executed;
- **Calculation engine** – used when Calculation Views or Analytic Views with calculated columns are queried.

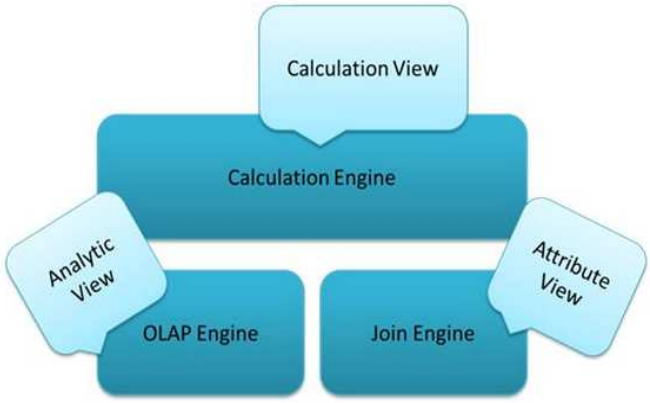

Fig. 4 Processing engines in HANA

A functionality implemented in HANA and frequently used in BI reporting is **the capacity of processing geospatial data**. A report revealing the nearest delivery point to the shop or which shop doesn't have enough storing space to assure the stocks are very important and is definitely more suggestive if this information is presented on a map. In this respect, in HANA there is a special engine called the *Spatial Engine* that is designed to support geospatial data and is integrated as an extension of the Calculation engine. The geospatial data can be accessed and operated through a standard set of methods [6].

The text analysis capacity implemented in HANA is designed to meet the actual challenges of the companies. Synthetizing or extracting the relevant and important information from the immense and various amount of data a company faces, is an

extremely important and useful functionality HANA owns [6].

There are two types of analysing the text [6]:

- "**Fuzzy search**" – allows searching of strings that only follow a defined pattern. For example, if we have a table designed for storing city names and we wish to search for "Munchen" in the specified column, there will be returned also the records storing names like "Munich", "Munhcen", "Munhen" etc.;
- "**Text Analysis**" – offers the option of complex text analysis on various strings, by defining some analysis rules for different industries and various languages. This allows *extraction of important information from unstructured data*, such as:

  - ***identifying parts of speech*** (nouns, verbs etc.);

  -named entity recognition: locations, persons, currency, dates etc.;

  . ***sentiment analysis,*** meaning that HANA offers the possibility to analyse a text and extract information that suggest if the communicated feelings are positive or negative.

In *Figure 5 –Text Processing in HANA* we can observe the entire flow of text processing in HANA [6]. Therefore, starting with an unstructured text data source and using dedicated algorithms, HANA analyses and extracts the most important features from the text. The result is afterwards stored in a table, as a structured text and can be used in reporting or analyses [6].

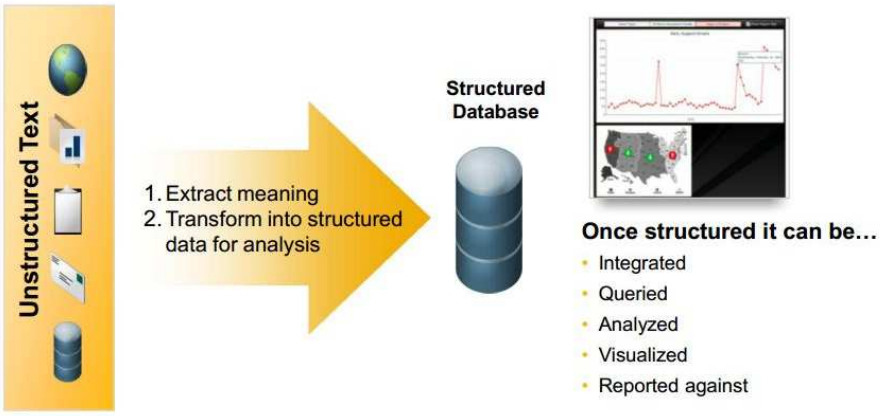

Fig.5. Text Processing in HANA

The Business Intelligence tools, as presented in *Figure 6 – BI tools* [6], can use the result of different types of data modeling available in HANA. In order to improve the quality of the querying process, the dedicated BI tools have been optimized to *push for execution at database level most of the complex operations*. This ways is promoted the concept of doing a small amount of operations at client level (application), many of them being optimized for execution at database level.

HANA data models can be a data source for the following tools, some of them not necessarily being specialized for the Business Intelligence domain:

- *Microsoft Excel* – the connection is made through ODBC connectors;
- *SAP Crystal Reports* – specialized tool used for "pixel perfect" designing of the reports;
- *SAP Business Objects Explorer* – allows exploring data in a fast way; it offers advanced options for viewing the data, providing also various graphic types;
- *Analysis for Office* – is an add-on installed on Microsoft Excel that allows data displaying as a datasheet and it offers special options for viewing the data, filtering and quick accessing;

- *Analysis for OLAP* – used for analysing and displaying the data in a Web environment;
- *Web Intelligence* – tool used for creating Web, Desktop or Mobile reports;
- *SAP Business Objects Dashboards* – used for creating dashboards.

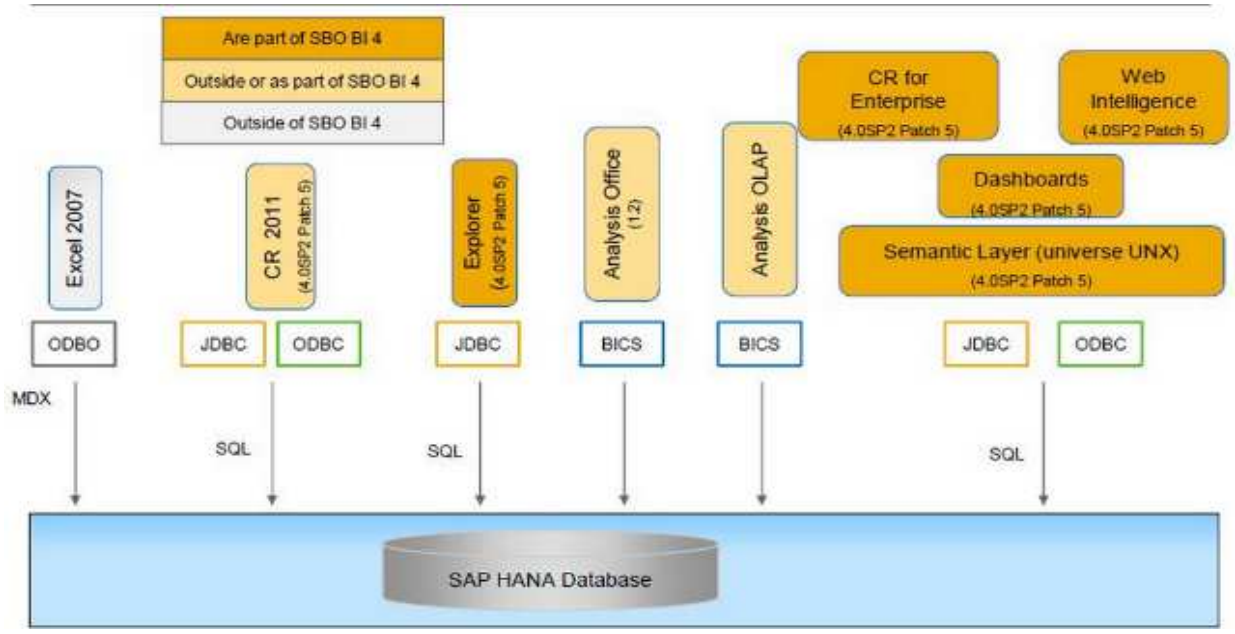

Fig. 6. BI Tools

6. Conclusions

In-memory databases are a technology that can bring tremendous value in a company. We must outline that this DBMSs are *column oriented*, assuring a high level of data compression this way. Additionally, *for indexing it is used a special algorithm*, usually based on T-tree. These systems offer **high performance**, even though they imply certain risks regarding the loss of data. There are many mechanisms that can be implemented in order to assure a safe recovering of data, such as: *high-availability* implementations, usage of *backup* copies and *journal* files etc.

SAP HANA is a top technology in this area and innovates the BI domain by bringing a variety of data modeling options and fast data processing.

References

[1] "In-memory database -Unabridged Guide" Walter Jennings, 2012

[2] "Main memory database systems: an overview" http://pages.cs.wisc.edu/~jhuang/qual/main-memory-db-overview.pdf

[3] "List of in-memory databases" http://en.wikipedia.org/wiki/List_of_in-memory_databases

[4] "In- memory database" http://en.wikipedia.org/wiki/In-memory_database

[5] "HA100 - SAP HANA Introduction", SAP SE (training material)

[6] "HA300 – SAP HANA Implementation and Modeling", SAP SE (training material)

[7] "An insight into SAP HANA Architecture" http://sapHANAtutorial.com/an-insight-into-sap-HANA-architecture/

[8] "Understanding SAP HANA Engine" http://sapHANAtutorial.com/sap-HANA-engine/

Ruxandra BĂBEANU graduated from the Faculty of Cybernetics, Statistics and Economic Informatics of the Bucharest University of Economic Studies in 2013. She is a currently graduating from the master program Databases-Business support at the Bucharest University of Economic Studies.

Marian CIOBANU graduated from the Faculty of Cybernetics, Statistics and Economic Informatics of the Bucharest University of Economic Studies in 2013. He is a currently graduating from the master program Databases – Business support at the Bucharest University of Economic Studies.