

# Building an Experiment Baseline in Migration Process from SQL Databases to Column Oriented No-SQL Databases

Gomez A<sup>1</sup>, Ouanouki R<sup>2</sup>, April A<sup>3</sup> and Abran A<sup>4</sup>

<sup>1</sup>PhD student-École de Technologie Supérieure (ÉTS) 1100, rue Notre-Dame Ouest (angle Peel) Montréal (Québec) H3C 1K3 514 396-8800 Bureau A-3438, Canada

<sup>2</sup>PhD student -École de Technologie Supérieure (ÉTS) 1100, rue Notre-Dame Ouest (angle Peel) Montréal (Québec) H3C 1K3, Canada

<sup>3</sup>Professor-École de Technologie Supérieure (ÉTS) 1100, rue Notre-Dame Ouest (angle Peel) Montréal (Québec) H3C 514 396-8800 Bureau A-3482, Canada

<sup>4</sup>Abran A, Professor-École de Technologie Supérieure (ÉTS) 1100, rue Notre-Dame Ouest (angle Peel) Montréal (Québec) H3C 1K3 514 396-8800 Bureau A-3482, Canada

## Abstract

Today No-SQL databases are becoming more popular in addressing Big Data problems to complement relational databases used in cloud environments. Currently, the migration process from relational databases to No-SQL databases is not clear and mainly based on heuristics approaches such as the developers' experience or intuitive judgments. This paper is part of a research whose goal is to propose a set of guidelines to support this migration process. This paper presents an experiment designed to obtain a baseline that allows a comparison between two migration approaches: the traditional heuristic approach without use of any guidelines and an approach where guidelines are used to help with the conversion. The experiment showed that guidelines can reduce the level of difficulty of the migration process.

**Keywords:** Column oriented databases; No-SQL databases; Distributed databases; Software experimentation; Cloud computing

## Introduction

Research in recent years has increasingly focused on Cloud Computing and on how to profit from its potential, especially for its novel databases applications. The importance of Cloud Computing has been demonstrated by companies such as Amazon, Facebook, Google or Twitter and it is generally accepted that the data generated by these companies far exceed the capacities of current relational database applications. The interest for these technologies is now reaching also the mainstream corporations. Indeed, Cloud Computing has brought new developments and new challenges for software engineering because of the large amount of data it is starting to accumulate and analyze on a daily basis. These challenges include several data management challenges such as capture, storage, query, sharing, analysis, and visualization. Handling such large amount of data has led to the creation of a new research area referred to as “big data”. This increasing growth in popularity for big data goes hand-in-hand with the growing need for experimentation in software engineering. Indeed, in their study of research on software engineering Zerkowicz [1] presented a survey, based on 612 papers chosen from three renowned publications: IEEE Transactions on Software Engineering, IEEE Software magazine and the International Conference on Software Engineering. He concludes in observing that over 50% of the papers analyzed did not present any experimental validation. In other words these researchers hypothesize but they do not validate their hypothesis using any valid experimentation. Similarly, we observe an absence of experimental studies in the field of Cloud Computing databases.

Until a few years ago, companies had used massively the relational database technology (RDBMS) to deal with their impressive amount of data. However, different case studies published by Abadi [2] show that accessing petabytes of data efficiently using RDBMS, in the cloud, is very challenging and workaround solutions, like sharding, creates many other problems. The emerging NoSQL databases, want to addresses these challenges. Unfortunately little work has been done to explore migration from RDBMS to No-SQL. For instance, Chonxing [3] proposed some migration rules for a conversion to HBase. However, more experimentation is needed, to show that a one-size-fits-all solution is not possible and not all applications are good candidates for this migration [4-6]. To really help the NoSQL neophytes, which are currently RDBMS experts, in understanding the new nomenclature

and syntax guidelines and examples would help greatly. A first step towards reaching this goal is to develop a set of guidelines for the migration from RDBMS to NoSQL database.

This paper presents the design of an experiment to establish a baseline that allows a valid comparison between the migration process of a database from RDBMS to NoSQL. This paper reports on the experiment, by, RDBMS experts, to conduct an RDBMS to NoSQL migration without the use of any guidelines, namely, using a heuristic approach. Secondly, in a near future paper, a second group of RDBMS experts will conduct the same migration but using guidelines developed by our research team.

The paper is organized as follows. Section II presents the motivation of this research, including the problem statement, the research objectives and its context. Section III presents related work and our experiment design. Section IV reports the experiment execution and its preliminary findings. Finally, section V presents the conclusions and future work.

## Motivation

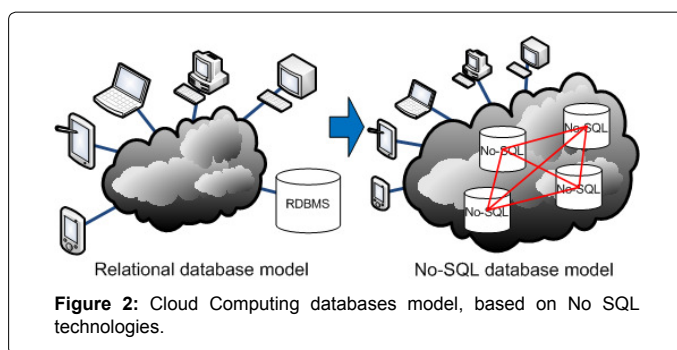
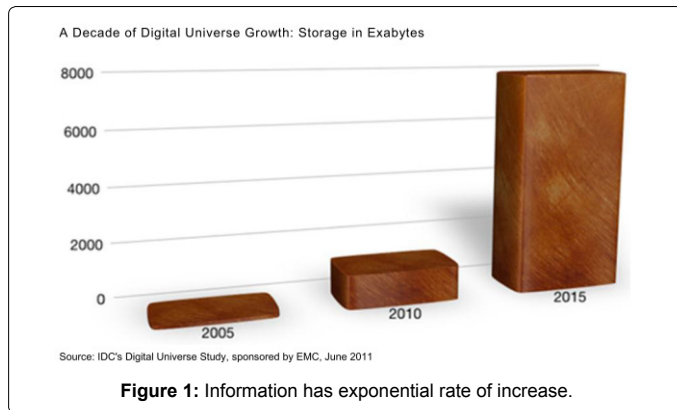
We are progressively all connected on the Internet. Consequently, many companies, regardless of which sector they operate in, have started capturing more and more information over the internet. Indeed, Internet has begun the cloud “*par excellence*”. Currently, organizations are making available countless numbers of software applications that use, on a large scale, relational databases (Figure 1). Information has exponential rate of increase. Terabyte (TB)=1000 GB. Petabyte (PB) 1000 TB. Exabyte (EB) 1000 PB. Figure 1 shows the current trend observed of the NoSQL database use, that is to say,

**\*Corresponding author:** Abraham Gomez, PhD student—École de Technologie Supérieure (ÉTS) 1100, rue Notre-Dame Ouest (angle Peel) Montréal (Québec) H3C 1K3, Canada, Tel: 514 396-8800; E-mail: [abraham-segundo.gomez.1@ens.etsmtl.ca](mailto:abraham-segundo.gomez.1@ens.etsmtl.ca)

**Received** October 29, 2014; **Accepted** December 20, 2014; **Published** December 30, 2014

**Citation:** Gomez A, Ouanouki R, April A, Abran A (2014) Building an Experiment Baseline in Migration Process from SQL Databases to Column Oriented No-SQL Databases. J Inform Tech Softw Eng 4: 137. doi:10.4172/2165-7866.1000137

**Copyright:** © 2014 Gomez A, et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.



by this year end the multiple of the unit byte for digital information normally used, is going to change from Terabytes to Petabytes, even Exabytes. This phenomenon was also predicted by John Gantz and David Reinsel [7] guarantees and storing space management [2]. That is, the administration of these systems becomes more and more complex, as reported by Figure 2 [2]. When a big data application that is using a relational database technology has shown its limits and that workaround solution, such as, sharing, are failing to solve the issues its time to think about No SQL technologies. Emerging Cloud Computing databases model, based on No SQL technologies, as shown in Figure 2, can address these challenges. Cloud computing solutions, whether public or private, along with the use of No SQL databases can, provide companies with new levels of economies of scale, agility, and flexibility compared with traditional IT environments based on the relational database model. It is predicted that, in the long term, these technologies will be a key toolset for dealing with the complexity observed in both, Cloud and big data applications.

#### A. Problem statement

Since the industry uses mainly relational databases and they are likely to migrate some of their large scale existing applications to a No SQL model, there is a research need to improve this process by identifying a set of guidelines to help database specialists in this first time migration from RDBMS to No SQL database. Our experiment will be focusing on an H Base migration, which is a popular column oriented No SQL database developed as part of Apache Hadoop project.

#### B. Research objectives

The current trend is moving the past local applications that use RDB to the Cloud (No-SQL) (Figure 3). One immediate consequence of this trend is that past local applications that use massively the relational database model are moving to applications in the Cloud. See Figure 2. Although, it is still very popular to use the relational

database model for Cloud applications, as shown in Figure 2, when the data deployed in the database servers (in the cloud) grow beyond 1TB, this technology starts to show its limits, e.g. in their work Stone braker [4,6,8] state the volume of data stored is related with problems in response time in the new research field called “big data”. Also the large increase in the number of users connected to cloud applications can cause other problems such as: transactional difficulties [2], ACID (Figure 3). This paper scope versus the overall research objective. Our research objective aims at investigating the benefits of the use of a set of guidelines as a way of improve the migration process of databases from RDBMS to NoSQL databases, focusing on HBase. In this article the concept “improve” is going to be used in the sense to bring into a more desirable or excellent condition the current migration process from SQL database applications to No-SQL database applications. Many authors have studied different ways to compare database application [9-12]. Figure 3 shows, graphically, our experimental research objective. We will use the same relational database application and follow two experimental tracks: the first, without the use of the guidelines, and the other one with the use of the guidelines. Many aspects have to be taken into account when migrate an existing application that uses an RDBMS database to a No SQL database. E.g. the evaluation implies the content evaluation of the two resulting database applications (which includes the coverage evaluation and the data correctness verification) and the SQL statement evaluation. At the end of the experimentation we will obtain two H Base databases and converted applications that can be compared (Figure 3). In order to conduct a comparison between the applications and resulting databases H Base and H Base’ we must conduct some preliminary experiments. The scope of this paper is to present these preliminary experiment results that serve as a baseline for the final research results and thus compare the results between the two H Base databases.

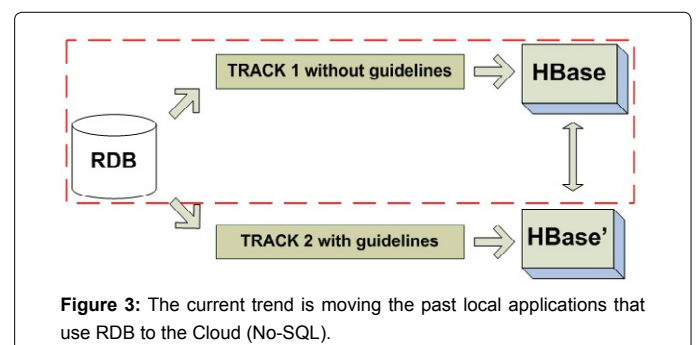
#### C. Context

The context and limitations of this research can be summarized as follows:

- The research results can be applied by any developer who wants to migrate an existing RBDMS application to column oriented No SQL databases.
- Unfortunately, we will see that not all the database applications are good candidates to be converted.
- In the same way, the results of this experiment cannot be extended to all No SQL database models, only to column oriented model, e.g, H Base, Cassandra or Hyper table.

#### Related Work

Few researches have addressed the problem of migration of a database application from relational environment to No SQL. Indeed,



there is very little information on how to do this, at least in a standard way; the options currently available to accomplish this migration are mostly based on an heuristic approach. It means, based on the developers' experience, educated guess, intuitive judgment, or common sense. This approach does not guarantee that an optimal solution will be found, but if it is properly done, it can provide a satisfactory solution. Salmen has proposed, in his initial attempt focused on identifying some of the core activities that are common to every migration process, to draw some general conclusions about how start this migration process (e.g. the DDI methodology proposed by Salmen [13], where DDI stands for Denormalization, Duplication, and Intelligent keys. Denormalization is the process used to optimize the read performance of a database by adding redundant data or by regrouping data. Data duplication can be defined as the occurrence of redundant data within a system by various means. An intelligent key is a database key which depends wholly on one or more other columns in the same table. An intelligent key might be identified for implementation convenience, when there is no good candidate key. Another contribution to this research field was published by Chongxin who developed some rules to help in the migration from relational databases to a specific cloud computing database, which is H Base [3]. However Chongxin explored a reduced set of ideas that he called "rules" (three to be precise) and this rules does not cover the entire characteristic that implies a relational database application today. Besides, Chongxin established their rules in a consecutive order, it means, in the first stage, I must apply the rule number 1, then the rule number 2 and finally the rule number three. This way to work significantly reduce the results of the method used. The last work was suggested by Singh in [14,15] In their work some general guidelines were proposed, but the problem is the guidelines were developed using the methodology of use cases that follow a heuristic approach and reduce the possibilities to replicate the work or adapted in general ways to applied in other contexts.

## Experimental Design

The experimental design was based on Jedlitschka's work [16]. As stated earlier (Section II, subsection B), this experiment is the first part of the entire research project to address the migration problem from relational to NoSQL databases. Only the results of the experimentation of track 1 will be presented and used as baseline for the future comparison with the track 2 experimentation results.

## Goals, Hypotheses, Parameters, and Variables

The experiment goal is to create a baseline that allows a valid comparison between the migration process of a database from relational database to NoSQL database, without the use of guidelines (heuristic approach) and with the use of guidelines (the proposed solution). The null hypothesis is:  $H_0$ : "there is no real improvement in the migration process with the use of guidelines; if there is any advantage, it is only coincidental, and the best option is to use an heuristic approach based on the developer's experience". On the other hand, the alternative hypothesis is:  $H_1$ : "there is a significant improvement in the migration process with the use of guidelines; this is not coincidental and the better option to achieve this process is to use the guidelines".

The experiment parameters are based on the research of Juristo and Wohlin [17,18]. The parameters are the invariable characteristics throughout the conduct of the experiment that do not influence their results. For this first experiment these parameters are: The experiment process, because it will be the same during both experiments (Figure 3). The migrated database characteristic such as tables, primaries keys, indexes, relationships or store procedures and that were used

to analyze the two resulting databases. The SQL statements used to compare the No-SQL databases, because it will be the same for both HBase databases. HBase, which is the specific No-SQL database selected for this experiment, for both experiment. The response variables (i.e. dependent variables) are related with the improvement aspects. First, the content evaluation will be analyzed. This can be subdivided into two separate analyses: 1) coverage analysis, and 2) data correctness analysis. The coverage analysis will compare the two HBase database based on a series of relational database characteristics found in the migrated HBase database: Tables, Fields, Relationships, Views, Indexes, Procedures, Functions, and Triggers. We will analyze the percentage, in coverage, of these characteristics in the resulting H Base solution presented once migrated. The data correctness analysis pertains to the data represented in the resulting H Base database: analyze the percentage of data correctness presented in the migrated database as compared with the original relational data base. In a second set of results analysis, the SQL statement will be analyzed based on their response time. Finally, the total amount of effort required to conduct the migration process will be captured.

The factors (called also independent variables or predictor variables) are also related with the participant's current level of expertise in both SQL and No-SQL: no expertise, intermediate, and advanced. In addition, the experimental case study application and database used in the experiment is of a small size and low-elaboration. Here the term "low-elaboration" is used to describe a database application that involves a high percentage of the main characteristic of database theory like tables, primary keys, store procedures and relationships but the calculated effort to build it is estimated between 20 and 30 hours of programming.

## Case study participants

This experiment was designed using the point of view of a typical developer. Taking this point of view, the participants are asked to state their experience level and are classified according to 1) their academic background 2) working field; 3) number of years of work experience with relational database, and 4) the number of years of work experience with any NoSQL database.

The word "experience" will be related to the domains of programmer, relational database programmer or relational database administrator. Moreover, the classification can be summarized according to different options. The academic background has the options Graduate with PhD, Graduate with Master, Graduate, and Undergraduate Student. The working field has the options Industry, Academic, and Research Center. The number of years of work experience with relational database environment has the options of No Experience, Low Experience (less than a year), Middle Experience (2 to 5 Years), and Advanced Experience (more than 5 Years). Experience in No-SQL database (Figure 4).

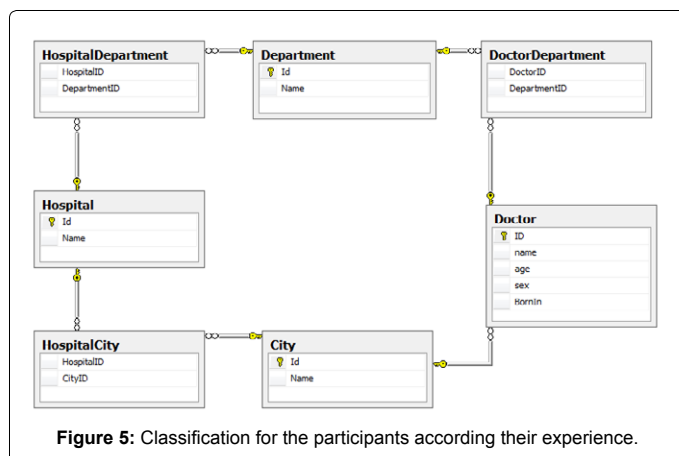
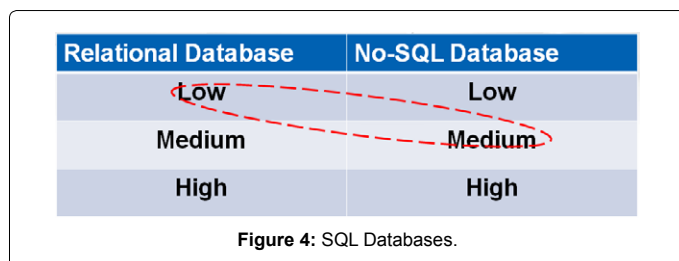
## Objects

The objects used in the experiment are as follows:

First of all, the synthetic database (the database source) given like a relational schema with a small size and low elaboration. It contains all the information in the form of fields, tables, views and relationships.

In the second place, the document with a similar example of migration process from relational database model to No-SQL database model. The document contains only an example and not the guidelines themselves.





## Instrumentation

In their study, Jedlitschka [16], describes this section like the appropriate to provide all information about the instrumentation that might have an impact on the results. There are two types of instruments: guidelines and measurement instruments. This experiment will use both of them. Concerning the guidelines, the participants will receive a package, in a yellow envelope, containing a series of documents: Firstly, the participants will receive several yellow pages with the training document, including relational database and Not-SQL explanations. Second, it will be given one blue sheet, with the synthetic relational schema that will be migrated to No-SQL: as previously noted, this schema has a small size and low elaboration. The schema is based on the research of Singh [14] and it can be seen in Figure 5 Classification for the participants according to their experience. The number of years of work experience related to any No-SQL database has the same options as above. The goal here is obtain a classification for the participants according to their experience that allows us to know the combinations (pair) relational-NoSQL experience that needs the solution and where it can be most useful. The Figure 4, for instance, highlight the pair Low-Medium, meaning a “low” experience in relational database environment and “medium” Relational schema given to the participants (Figure 5).

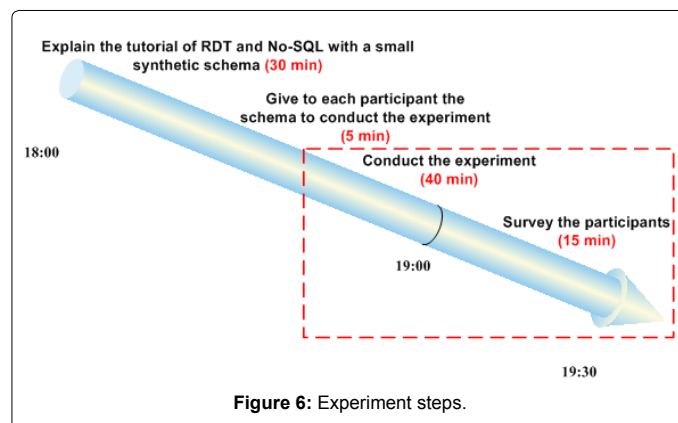
It is important to note that the participants will have the option to choose the sub-schemas that they will select for the migration. For instance, one participant can choose only convert the sub-schema composed by the entities Hospital-Hospital City-City or he can select the entire schema. In the third place, it will be given one green sheet where the participant will write or design the No-SQL solution. In addition, the participant will receive several white sheets that he can use as drafts. Lastly and with regards to measurement instruments, a survey will be applied to the participants, after the experiment. The Figure 6 summarizes all the experiment steps. The red dotted line indicates the experiment itself, with the expected time of each activity.

## Data collection procedure

Easterbrooke, Marcos, and Zelkowitz [1,19,20] describe this section as the right place for presenting the details of the collection method, from manual collection by the participants to automatic collection by tools. In this experiment the data collection procedure was manual because each participant received a schema to conduct the experiment. Besides, the procedure was conducted inside the process indicated by the red dotted line of Figure 6. Despite that this experiment does not cover the guidelines; it is important that the synthetic relational database used in the experiment has multiple examples of each guideline, and at least one. Another manual procedure was the survey, which was designed following the research work devised by Kasunic and Lethbridge [21,22]. It was composed of nine questions, with the first four were totally oriented to “*experience classification*”, as explained in participants sub-section. The fifth question was related with the migration process and the opinion about the first step to begin it. The sixth question was related with the effort needed to achieve the process without the guidelines. This question was rated from 1 to 5, where 1 indicates that the process was easy to achieve without effort, a value of 3 indicates that it was required a maximum effort to achieve it and a value of 5 means that no matter how comprehensive the effort, it was not possible to achieve it. The seventh question was designed to evaluate their level of confusion during the process, e.g. no idea where to start or what the next step was. The questions were rated from 1 to 5, where: always confused, very often confused, sometimes confused, rarely confused, and never confused. The eighth question is a matrix for evaluating the percentage that covers the designed solution with regards to the relational aspects mentioned earlier (Table, Constraint, PK, and FK). Finally, the ninth question, the participant’s opinion to know if he/she thinks that to receive some guidelines could improve their process. This question was rated 1 to 5 with the levels: strongly agree, agree, undecided, disagree, and strongly disagree.

## Conduct the Experiment Design

In the experiment execution has participated eighteen colleagues: Twelve participants belong to the industrial sector and four participants were graduate students at the École de Technologie Supérieure (ÉTS). All participants were provided with a clear and well established knowledge about the purpose of the experiment. The material used in the execution was: The document including the call for participants. The participant’s instructions. The final survey form. The participants training document (White document). It was a document that summarizes the training part explained at the beginning of the experiment. The synthetic relational schema (Blue document). This was the schema that must be migrated to No-SQL. The No- SQL solution



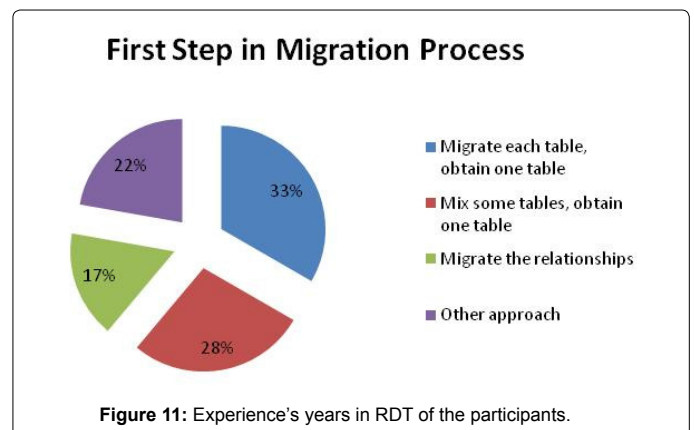
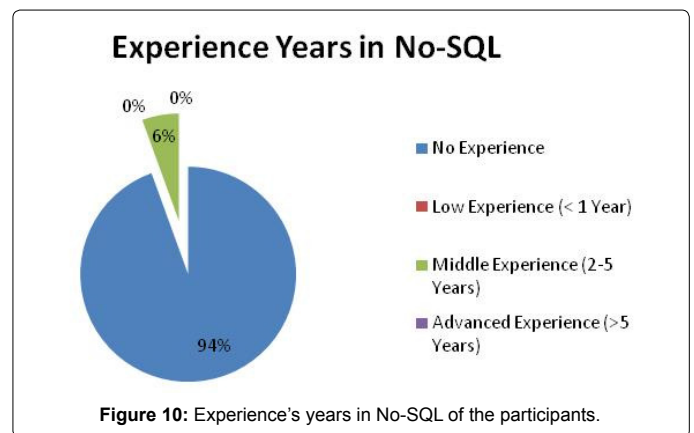
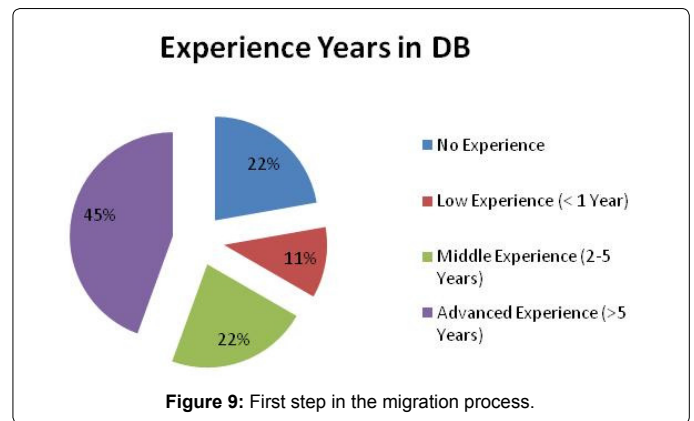
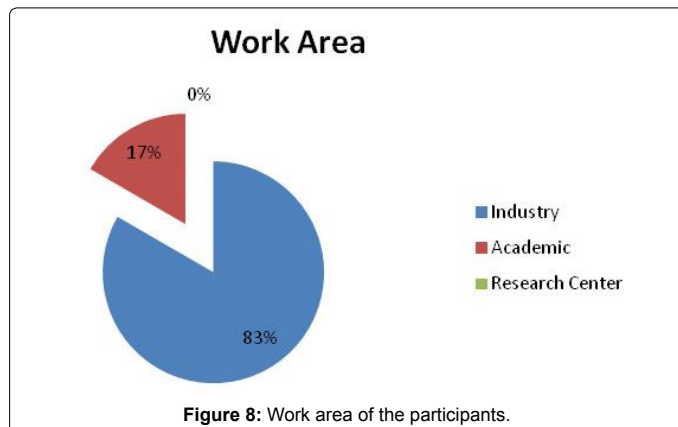
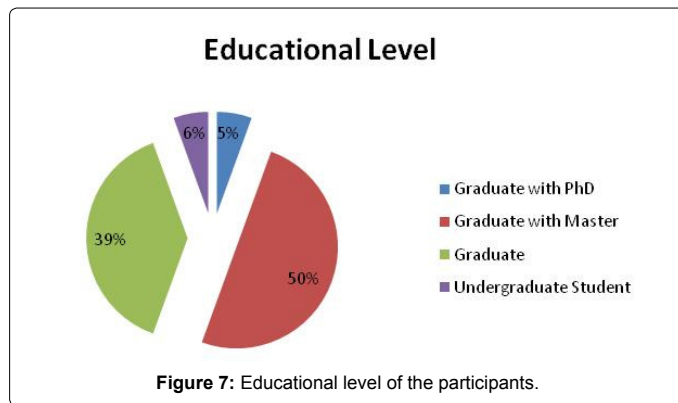
(Green document). This was an empty sheet, where the participant could draw the new schema resulting from their knowledge.

The drafts documents (Yellow documents). It means sheets to draw any thing the participant could use as support.

## Case Study Results

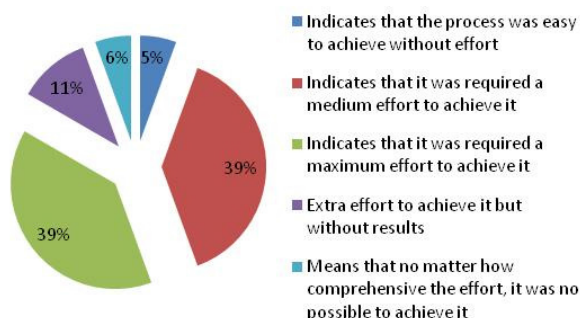
As mentioned previously, there are no experiments and data that support conclusions or decisions in the domain of migration from RDT to No-SQL databases. Generally speaking, all the migrations have been conducted using an heuristic approach, e.g., the developers experience or the developer's educated guesses or their common sense. The goal of this paper is using the results obtained as a baseline for comparisons in future stages of the entire research. The experiment process, presented in Figure 6, consisted of two well established parts, first at all an explanation of all the technological context, it means, a tutorial about the RDT and the NO-SQL technology, a duration of 30 minutes was scheduled. After, all the participants received the documentation stated in section V. Subsequently the participants conduct the experiment, eventually filling the green sheet (the No-SQL schema resulting from the migration). Finally they expressed their opinions filling a survey (Figure 7). Educational level of the participants. Figure 7 indicate a low level of interest from undergraduate students to participate in this kind of studies. Besides, in the participants is found an 89% of graduate that shows an interest to conduct the experiment. (50% graduates with master plus 39% of graduates). Figure 8 evidence a great participation the participants that chosen to begin with the tables. This leads from industry sector (83%) to think that start by the tables could be a good guess (Figure 8). Work area of the participants.

It can be observed in Figure 9 that a great number of participants



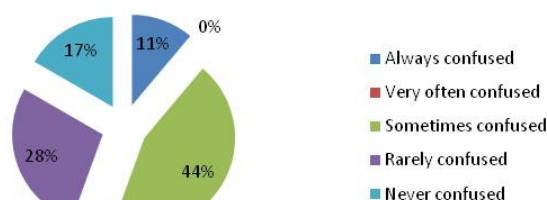
have the experience in RDT field. A 45% have more than 5 years of experience and this result together with the Figure 8 result (83% of participants in industry sector) give a lot of value to the results of this experimentation. The difficulty during the whole process is reports by the Figure 12. As can be seen, the initial perception that the procedure is difficult was unchanged (near 78% resulting from 39% plus 39%). This notion was reinforced considering the Figures 9 and 10. First step in the migration process. So the participants think the process demands a considerable amount of effort, because the No-SQL databases are totally new for them (Figure 9). Experience's years in RDT of the participants. In contrast, Figure 10 illustrate that a 94% of the participants have no knowledge about No-SQL databases technology. The results show by Figures 10 and 11 strongly indicates that a set of guidelines could be an invaluable tool for the RDT experts in migration process. Experience's

### Level of Difficulty in Migration



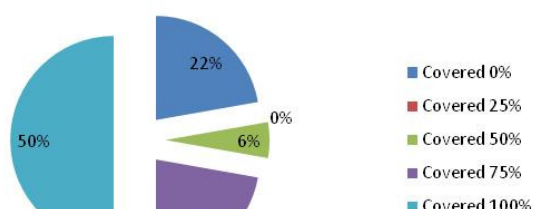
**Figure 12:** Level of difficulty in the migration process.

### How to Begin? What Is the Next Step?



**Figure 13:** How to begin the process.

### Criteria "Tables" was Covered



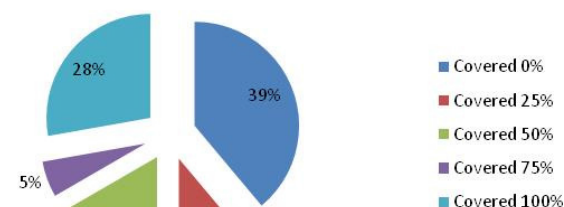
**Figure 14:** Tables covered.

years in No-SQL of the participants (Figure 12). Level of difficulty in the migration process. Following on from the foregoing, the Figure 13 demonstrates that the majority of the participants (44%) felt themselves sometimes confused, i.e., without knowing how to go about it. How to begin the process. As regards the first thing to do at the beginning of migration process, the Figure 9 provides the different paths presented in the participants. Considering Figure 8 (83% in industry sector) and Figure 11 (45% with more than 5 years of experience), there was a large proportion 61% (resulting from 33% plus 28%).

In matters of the improvement aspect considered for the relational databases, for the experiment has studied only five: tables, constraint. The Figure 14 reveals that 50% of the participants think that their solution cover the "tables" aspect in a 100%. In contrast 22% think that their solution covered this aspect in a 0% (Figure 14). The Figure 15

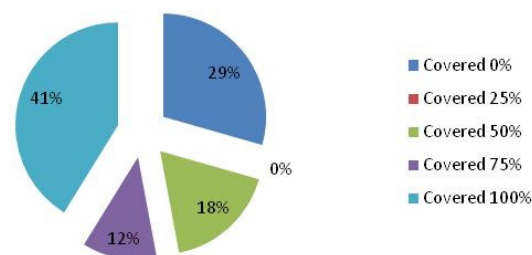
presents that 28% of the participants think that their solution covers the "constraints" aspect in a 100%. In contrast 39% think that their solution covered this aspect in a 0%. However, as was stated in section IV, subsection A, the synthetic database used in the experiment is of small size and low elaboration, i.e., it was conceived without the presence of constraints, so this is an important thing to consider (Figure 15). Constraint covered. The Figure 16 reports that 41% of the participants think that their solution covers the "primary keys" aspect in a 100%. In contrast 29% think that their solution covered this aspect in a 0% (Figure 16). Primary Keys covered. The Figure 17 shows that 39% of the participants think that their solution covers the "foreign keys" aspect in a 100%. In contrast 28% think that their solution covered this aspect in a 0% (Figure 17). Foreign key covered other relational

### Criteria "Constraint" was Covered



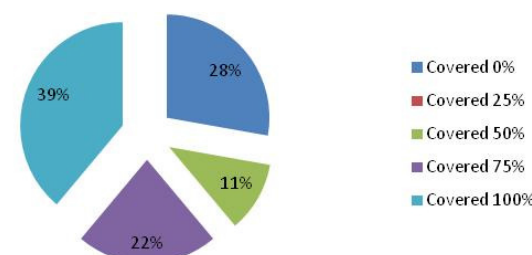
**Figure 15:** Constraint covered.

### Criteria "Primary Key" was Covered



**Figure 16:** Primary Keys covered.

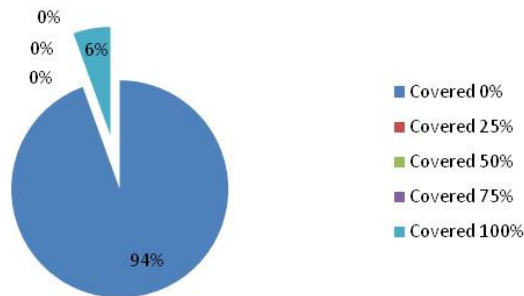
### Criteria "Foreign Key" was Covered



**Figure 17:** Foreign key coveredOther relational database improvement aspect like fields.

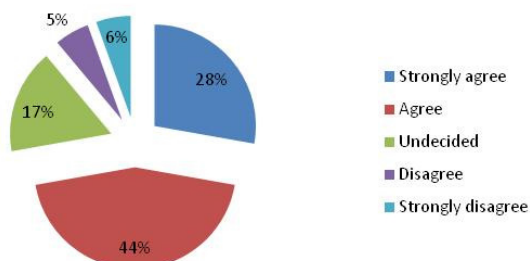


### Criteria "Others" was Covered



**Figure 18:** Others criteria covered.

### Guidelines Evaluation



**Figure 19:** Guidelines evaluation.

database improvement aspect like fields, store procedures or triggers were put together in the aspect "others" and the Figure 18 reveals that 94% of the participants show no interest in these aspects (Figure 18). Others criteria covered finally the Figure 19 provides the opinion of the participants in case that a set of guidelines it had been provided. 28% are strongly agreed about their usefulness and 44% are agreeing with the relevance of this kind of tool in the migration process (Figure 19).

### Analysis and Interpretation

Fortunately, it is possible to show some feedback based on comments received during the workshop. Any information about guidelines was given to participants. It is reasonable to assume that those without familiarity in database have experienced more difficulties than others with some years of working with them.

The comments about the training session were positive in general. Despite the experiment trainer's effort, it can be observed that during the first half hour of the experiment there was a considerably spent of time consulting the reference documentation, especially those participants without the requested experience. According the feedback of some PhD students, the first obstacle was to figure out what could be the first step to start the process. However, it is necessary to wait until the results are treated properly and appropriately. We expect to complete the analysis of the data by December 2014.

### References

1. Zekowit MV, Wallace DR, Binkley DW (2003) Experimental validation of new software technology, in Lecture notes on empirical software engineering, World Scientific Publishing Co, Inc. p. 229-263.
2. Abadi D (2009) Data Management in the Cloud: Limitations and Opportunities. IEEE Data Engineering Bulletin, Vol 32.

3. Chongxin L (2010) Transforming relational database into H Base: A case study. in Software Engineering and Service Sciences (ICSESS), IEEE International Conference on pp 683-687.
4. Stone braker M, Madden S, Abadi DJ, Harizopoulos S, Hachem N, et al. (2007) The end of an architectural era: (it's time for a complete rewrite), in Proceedings of the 33rd international conference on Very large data bases., VLDB Endowment: Vienna, Austria. p. 1150-1160.
5. Stonebraker M, Abadi D, DeWitt DJ, Madden S, Paulson E, et al. (2010) Map Reduce and parallel DBMSs: friends or foes? Commun ACM 53: 64-71.
6. Stonebraker M (2008) Technical perspective: One size fits all: an idea whose time has come and gone. Commun ACM 51: 76-76.
7. Gantz, J, Reinsel D (2011) Extracting Value from Chaos, in EMC Corporation, E. Corporation, Editor. p. 12.
8. Stone braker M (1986) The Case for Shared Nothing. A quarterly bulletin of the IEEE computer society technical committee on Database Engineering. Vol 9: p. 6.
9. Boral H, DeWitt DJ (1984) A methodology for database system performance evaluation. SIGMOD Rec. 14: 176-185.
10. Varia J (2010) Migrating your Existing Applications to the AWS Cloud, Amazon Web Services.
11. Demarest G, Wang R (2012) Oracle Cloud Computing, Oracle, Editor, Oracle: Redwood Shores, CA.
12. Annamalai M, Oracle (2012) High Performance Connectors for Load and Access of Data from Hadoop to Oracle Database. 20.
13. Salmen D, Malyuta T, Feters R (2009) Cloud Data Structure Diagramming Techniques and Design Patterns.
14. Singh P (2010) Schema Guidelines & Case Studies.
15. Singh T, Sandhu PS (2011) Cloud computing databases: Latest trends and architectural concepts. Proceedings of World Academy of Science, Engineering and Technology, 73 (Compendex): p. 1042-1045.
16. Jedlitschka A, Pfahl D (2005) Reporting Guidelines for Controlled Experiments in Software Engineering. In Proceedings of the 4th International Symposium on Empirical Software Engineering (ISESE) IEEE Computer Society.
17. Juristo N, Moreno AM (2010) Basics of Software Engineering Experimentation. 2010: Springer Publishing Company, Incorporated.
18. Wohlin C, Höst M, Henningsson K (2003) Empirical Research Methods in Software Engineering, in Empirical Methods and Studies in Software Engineering, R. Conradi and A. Wang (Eds), Springer Berlin Heidelberg. p. 7-23.
19. Easterbrook S (2008) Selecting Empirical Methods for Software Engineering Research Guide to Advanced Empirical Software Engineering, in Guide to Advanced Empirical Software Engineering, F. Shull, J. Singer, and D. Sjøberg, Eds. Springer London. p. 285-311.
20. Marcos E (2005) Software engineering research versus software development. SIGSOFT Soft w. Eng. Notes, 30: 1-7.
21. Kasunic M (2005) Designing an Effective Survey.
22. Lethbridge TC (1998) A Survey of the Relevance of Computer Science and Software Engineering Education, in Proceedings of the 11th Conference on Software Engineering Education and Training, IEEE Computer Society. p. 0056.