Proceedings of the International Conference on Computer and Information Science and Technology
Ottawa, Ontario, Canada, May 11 – 12, 2015
Paper No. 105

Predicting the Impact of the Factors That Influence the Adoption of Multi-Tenant Databases

Olumuyiwa Matthew, Mary Garvey, Kevan Buckley
University of Wolverhampton,
Wulfruna St, Wolverhampton, West Midlands,
WV1 1LY, United Kingdom
o.o.matthew@wlv.ac.uk; M.Garvey@wlv.ac.uk; K.A.Buckley@wlv.ac.uk

Abstract– Considering the extremely low utilization of dedicated database servers and also reducing the hardware and operational cost associated with this model of database deployment, it is more reasonable to adopt a multi-tenant database (MTD). A MTD is a way of deploying a Database as a Service (DaaS). This is gaining momentum with significant increase in the number of organizations ready to take advantage of the technology. A multi-tenant database refers to a principle where a single instance of a Database Management System (DBMS) runs on a server, serving multiple clients organizations (tenants). This is a database which provides database support to a number of separate and distinct groups of users or tenants. This technology amortises the total cost of ownership (TCO) thereby leading to reduction in per- tenant cost by increasing the scale. This paper considers the requirements and challenges of implementing MTDs. Several factors that influence the adoption of MTDs are identified. In this work, a research survey is presented that use a questionnaire while UK Oracle user group (UKOUG) members are the focus group. Thirty respondents' data were collated and carefully analysed. The primary contribution of this paper is to present to intended subscriber of MTD the degree of impact these factors has on the decision to adopt any of the MTD model.

Keywords: Databases; Deployment; Multi-tenant; Momentum; Amortise; Tenant.

1. Introduction

In today's world of business, companies' vital and sensitive information are accessed, stored and transferred electronically. The security of this information and its storage system are critical to the reputation and progress of the companies. Therefore, the method of storage and accessibility of this information guarantees safety.

A MTD refers to a principle where a single instance of a Database Management System (DBMS) runs on a server, serving multiple clients (tenants). Multi-tenant database is one which provides database support to a number of separate and distinct group of users, also referred to as tenants. A tenant is simply any logically defined group of users that requires access to its own set of data. This definition was substantiated by Bezemer et al (2010 p1) as an architectural pattern in which a single instance of the software is run on the service provider's infrastructure, and multiple tenants access the same instance. This reduces effort made in production and the cost incurred in the development. In a multi-tenant enabled service environment, user requests from different organizations and companies (*tenants*) are served concurrently by one or more hosted application instances and databases based on a scalable, shared hardware and software infrastructure (Gao et al 2010 p324). Such database systems must be able to maintain or even increase their performance or efficiency level under a larger operational demands. The concept of multi-tenancy was developed from the service providing technology known as Software as a Service (SaaS). SaaS is a form of cloud computing that involves offering software services in an on-line and on-demand fashion with the Internet as the delivery mechanism (Walraven et al 2014).

Organisations incur huge cost on the acquiring and maintaining dedicated database system which ranges from cost of infrastructures, software licences, maintenance, monitoring, managing and upgrading. The adoption of MTD will eliminate most of these cost. However, there are some important factors needed to be considered before the MTD adoption. This paper presents these factors and also examine the degree of impact each of these factors could have on the adoption of the MTD concept.

2. Literature Review

SaaS constitutes a fast-growing business model for the sales of software that is based on the principle of outsourcing. With SaaS, a service provider hosts an application or software on its infrastructure and delivers it as a service to several tenants. These tenants subscribe for the service and accesses it across the Internet through standard web technology (Schiller et al 2011 p117). In the same way general services and DBMS service can also be out sourced for some reasons that are obvious to the tenants.

According to Curino et al (2011) in Ni et al (2014 p2079) most traditional enterprises' databases are deployed on dedicated database servers. Usually these servers are not fully utilized during much of the time. Curino reported that in traces from almost 200 production servers from different organizations, an average CPU utilization is less than 4 percent. The extreme low utilization can be solved by consolidating multiple databases on one or fewer machines, reducing hardware and operational costs. Thereby making multi-tenant data management system to serve a way of amortizing the cost of hardware, software and professional services to a large number of tenants and thereby leading to reduction in per-tenant cost by increasing the scale. This means that the more tenants you have on the server the less the CPU experiences idle time. As the service provider wants to support as many tenants as possible, the multi-tenant database system requires to have excellent performance, low space requirement and good scalability (Ni et al 2014).

2. 1. Requirement of Multi-Tenant Databases

A multi-tenant database should maintain an instance of a base schema for each customer. The Structured query language should ensure that DDL statements for modifying the base schema and DML statements for transforming existing data within it are applied to all customers in the farm, within the context of a rolling upgrade. The ability to perform such operations in bulk on the individual databases is essential to minimize downtime during an upgrade (Jacob and Aulbach 2007 p2). DML are SQL statements that involve modifying data which include inserting, updating or deleting rows while DDL are SQL statements that involve defining database elements like defining data tables, views or indexes (Dekimpe et al 2002).

Some of these requirements are also emphasized by Gao et al (2011 pp324-325) which includes:

1. Low Delivery and Operation Cost - Lower the cost of Hardware, software and utility of hosting center (bandwidth, power, space etc.) Lower the cost of human resources to maintain the processes and lifecycle via optimization and automation.
2. Easy and Low Development Cost – Granting ability to the developer to customize system for business application without bothering about the technicalities of the Multi-tenant issues.
3. Security isolation - Security Isolation refers to the mechanism where a user is prevented to obtain the priviledge to access resources belonging to other tenants. This is to ensure that each tenant is safeguarded and protected like in the traditional single-tenanted application.
4. Flexibility. The fundamental design point of SaaS is to serve hundreds and thousands of tenants through one instance of software application. But tenant usually has its own requirements such as specific object attribute, business logic etc.
    a. Customization - Dynamically extend the attributes of existing business objects or
    b. Diverse SLA. Provide multiple kinds of service options with different SLA (security level, concurrent users, data size, data encryption, backup period etc.).
5. Scalability and availability- Ability to scale very large to support very large customer volume; Incremental scalability, scale-out without impacting the service availability of other tenants.

2. 2. Challenges of Implementing Multi-Tenant Database

Ying et al (2011 p335) explained some of the challenges associated with multi-tenancy database development against the traditional database. The first challenge is the data isolation among tenants. The database must ensure the data of these tenants are isolated from each other and no one can access their data other than themselves. The second challenge is to achieve the economics of scales; the database must have the capability of on-demand scale to support large volumes of tenants. This means that irrespective of growth in number of tenants and their demand on the database, it must be capable of meeting the demand. Wood and Anderson (2011) argue that complexity through the different and changing demands and requirement of tenant raises further concerns in regards to maintaining and controlling the system. Because of this changing demand over time, the issue relating to scalability and security must be taking into consideration in deploying a multi-tenant environment. The third challenge is to be transparent to current existing application/skill, that is, the cloud developers can easily deploy the existing applications to on multi-tenant database without a large amount of code change, and the developers can create new multi-tenant application without using new technical knowledge. The forth challenge is to support different isolations for the same application. This means that the use of different applications by different tenant should ensure data isolation to each tenant regardless of the number of tenants involved.

These same constraints were also mentioned by Fang and Tong (2011 p95). Due to the peculiarity need of each tenant, there are problems of; 1, whether the database can afford the increase of both data and request accompanied with the growth of tenants. 2, how the database can meet the specific needs of one tenant efficiently and safely without affecting the others. It seems that the basic challenges associated with this technology since remain the same and different models or approaches were proposed to handle each of these challenges.

2. 3. Factors Influencing the Choice of Multi-Tenant Database Approaches

There are factors that help in determining the most suitable and appropriate approach of Multi-tenants database. The use of the system should be one of the influencing factors towards the decision. Elmore et al (2011 p5) emphasise that the tenant application and usage requirements should be the primary consideration in deciding the right model of multi-tenant database. Sometimes users (tenants) are not equipped with necessary information about this before taking decision on what approach to adopt. Their decision is sometimes influenced by what vendors tell them. There is need to examine all these basic factors before approaching a service provider in order to make the right decision on this.

Some of these factors are itemized by Keemti (2010) as follows.

1. Size of tenant database.
2. Number of tenants.
3. Number of users per tenant.
4. Growth rate of tenants.
5. Growth rate of tenant database.
6. Security.
7. Cost.
8. Flexibility – ability to create multiple tables by tenants.
9. Regulatory consideration (UK/EU countries)

Some of these factors were mentioned by different scholars depending on the context of their research. It is possible to group them into four major headings as follows:

- Cost consideration.
- Growth consideration.
- Security consideration.
- Regulations consideration.

2. 3. 1. Cost Consideration

Cost is vital when considering the appropriate approach to be adopted in the implementation of database multi-tenancy. This cost is referred to as total cost of ownership (TCO) which is broken down into three major types. These are infrastructural cost, management cost and application development cost (Wang et al 2008 pp94-95). Infrastructural cost includes the cost of hardware, software and utilization costs. And management cost are cost related to the operational activities and processes like lifecycle management, monitoring data backup and restore while application development cost are cost related to meeting each customer additional unique requirement (Wang et al 2008 pp94-95). Considering all these costs involved in acquiring a dedicated database system, it is obvious that huge amount of resources are involved therefore you might want to consider an approach that will require less cost of development than a large-scale development approach.

2. 3. 2. Growth Consideration

A decision was made to group all points 1-5 of the above mentioned factors as part of growth consideration factor. Aulbach and Jacobs (2007pp3-4) carried out experiment on memory (storage) and disk usage of the five different databases. This experiment focused on the shared process approach which is a model type of multi-tenant implementation. This shows that the size of tenant database in terms of storage capacity is a factor needed to be considered during the decision about choice of approach.

Schiller et al (2011 p118) give an illustration that the shared table approach has a promising for a service provider that has target for long tail because it offers the lowest over-head per tenant and thus suitable for a large number of small tenants. An example of 1000 tenants with each uses less than 50MB of data and at most 5 concurrent users. This is an indication that the number of tenants on the database and the number of users per tenant are all factors which must be thoroughly examined and also contribute to the performance of database system which invariably is also a factor to be considered when taking decision on what approach to be used. Myer (2007) in Schaffner (2013 p1) reveal that already in October 2007, the SaaS CRM vendor RightNow had 3,000 tenants which are distributed across 200 MySQL database instances with 1-100 tenants per instance. This means that the CRM can evolve overtime. It's been designed to accommodate more tenants. The growth rate of tenants on it is never limited to a small number.

2. 3. 3. Security Consideration

In multi-tenant database systems, one of the major concerns is the risk of data being exposed to the third parties. Based on the fact that every database is design to store sensitive data, the prospective tenant will have very high security expectations. Every service provider will always want to operate to much higher security standard but sometimes this might not be to a hundred per cent. Therefore the service level agreement (SLAs) will have to provide very strong data security guarantees.

Some of the security issues related to multi-tenancy database include data isolation among the tenants. Gao et al (2011 p324) mention some of the challenges facing the ISVs (Independent Service Vendor/Provider) in delivering their service these include the data security isolation among tenant, the different tenants having different SLA demand, customization requirement and effective database scale out mechanism as the number of tenant increases. Hui et al (2009 p832) itemises the problems faced in providing database as a service which includes security, contention for shared resources and extensibility. Hui et al (2009) also mentioned scalability as one of the problems which was defined as the ability to serve an increasing number of tenants without too much query performance degradation. In spite of the increase in the tenancy and query request, the system should still be able maintain its performance level. This tenancy growth should not impact the service availability of other tenants.

Another major security issue is the flexibility of the system. A Multi-tenant database must be able to serve hundreds and thousands of tenants in one instance. Aulbach et al (2009 p881) expand this by saying that a multi-tenant database must be flexible by being able to extend the base schema to support multiple specialised version of application and to dynamically evolve the base schema and its extension while the database is on line.

It was also pointed by Vashistha and Ahmed (2012 p49) that isolation should be carefully considered in almost all part of architecture design from both functional and non-functional level such as security, performance, availability, administration and also support tenant customization of their own services in runtime without impacting on others.

2. 3. 4. Regulatory Consideration

There are laws and regulations put in place by different governments that serve as protection to databases of different entities that operate in that geographical location. Companies, organisation and governments are often subject to regulatory laws that affect their security and even record storage needs. The knowledge of these different laws and regulations are considered when making decision about multi-tenant database. Chong et al (2006) argue that the investigation of regulatory environments that your prospective customers occupy in the market in which you expect to operate is important. This shows that you have to be conversant with the laws that operate in that area or country. It is important to find out whether there are any aspect of the law that present any condition that will influence your decision toward given your database service to a third party.

There are challenges when tenants are from different regulatory authority. Most of the countries have different laws which sometimes are not having same conditions and to now manage these conflicts will pose a great technical and economical challenge to the ISVs. Effort must be made to look into the regulatory provisions for prospective tenants. The regulatory laws of all the tenants on a particular multi-tenanted database must be the same in order to have harmony in the laws that govern the system.

All these are major consideration that must be considered to make a good decision about the adoption of a multi-tenant database model depending on the tenant individual system requirements. Therefore, this paper investigates different factors that influences the adoption of Multi-tenant database and to determine the level of impact of each of the identified factors in the consideration of the concept.

3. Data and Methods

A survey was carried out using quantitative approach where questionnaire were used as the survey instrument. One major reason for using a survey is that surveys are viewed as the most appropriate method of studying participants' behaviour and job perceptions (Mintzberg, 1973; Rea and Parker, 1997). According to Burns (2000); Creswell (2003) questionnaire is one of the best means of obtaining survey data. The questionnaire was designed carefully to make sure that it demonstrates and draws out useful responses. It was made easy to read and understand, short and have potency of being completed quickly. There are simple instructions given to the participants at the beginning of every section for easy, personal and independent inputs. A sample of the questionnaire is attached to the appendix. The data extracted from the questionnaire were carefully analysed with the use of a simple quantitative statistical method called weighted score method.

The UK Oracle user group (UKOUG) members were the participants in this survey. This is an association formed since 1983 with the mission to serve the Oracle community. This is based in United Kingdom and made up of over 8,500 people working for a variety of Oracle customers, Oracle partners and Oracle corporations. They come together every year with the purpose of education, innovation and information. The survey was conducted in the December, 2013 conference held in Manchester. For ethical reasons, all respondents' personal details were made anonymous. Thirty responses were collated and carefully analysed to obtain the outcome of this paper.

This group of people were selected because of their high level of involvement with databases and also because of the recent product of Oracle called 12C which is based on the multitenant concept.

4. Analysis

The graphical representation of the responses based on each factor are shown below. The graph shows the lines of the rankings in relation with each factor.

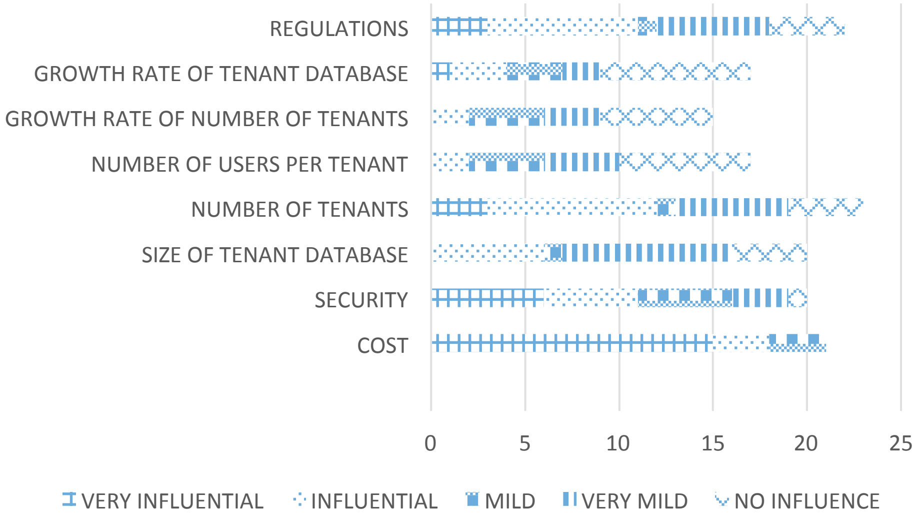

Fig. 1.

The graph above shows that cost has the highest value for very influential with 15 respondents agreed for cost as the most influential factor. And none of the respondent indicate that cost has no influence. The growth rate of number of tenants has the highest value for no influence with 8 respondent agreed that it is of no influence to the adoption of MTD. All the factors (1-5) above that makes the growth factor has values for no influence and when put together it shows that growth might has the least effect on the decision to adopt MTD.

In analysing this data further, a simple quantitative statistical method was adopted known as weighted score method also known as numerical indicator (Abeysekera 2001 p10). The results from the method are represented in the table 2 below.

Table 1. Initialtable of Results

Factor	Result
Cost	4.57
Security	3.6
Size of Tenant Database	2.45
Number of Tenants	3.04
Number of Users per Tenant	2.05
Growth rate of Number of Tenants	2.13
Growth rate of Tenant’s Database	2.24
Regulations	3.0

5. Findings and Discussion

The respondents’ data shows that some gave ranking to some of the factors while some factors were not ranked. This might be that some of these factors that were not ranked doesn’t have impact or influence on the decision towards MTD. However, looking at the result of the analysis, cost has the highest indicator value of 4.57 which means it has the greatest impact based on the perspective of the respondents. This means that cost should be first factor to be considered. This is the total cost of ownership which include several cost as explained above. Since every organisation is set up for business purpose that means that any form of saving cost and increasing revenue will inform the decision to be taking.

Security is another factor that has second indicator value of 3.6 and should be consider after cost has been taking into consideration. Data safety remains one major aspect any organisation would like to maintain. The risk of data being exposed to another party must be guided against in MTD so that perspective tenant will be confident with the service provided. There should be a good service level agreement to protect your data policies.

Regulations has a value of 3.0 which means that this is consider even before the growth factor. Regulations that governs the environment of your tenants as a service provider must be considered. Tenants should be conversant with the laws and regulations that protect your database before even considering the MTD service provider to engage. These regulations differs based on regions, continent, nations or state where tenants are located.

Growth which includes the following factors; the size of tenant database, number of tenants, number of users per tenant, growth rate of number of tenants and growth rate of tenant’s database has an average value of 2.38 shows that is the least factor that should be considered. This might suggest that it will have the least degree of impact on the decision about MTD.

6. Conclusion

This research has shown that MTD will enable service provider to transform a fixed service to a variable services of their technology footprint to enhance business agility, optimize their operations and lower their operational cost thus drive business. Also, this research has been able to identify the factors and proved that these identified factors has real impact on the adoption of MTD. These set of responses from the participants were able to prove the impact of these factors through their contribution in this research. However, this stage of the research has not proven whether it is positive or negative impact that is whether it drives towards the adoption or away from the adoption of the concept.

This research has also shown that MTD helps reduce the TCO involved in acquiring a dedicated database system. The growth of the MTDs include the number of tenants, the number of users per tenant, the size of each tenant's database, growth rate of tenants and the growth rate of tenant's database should all be a consideration in adopting MTD. Technical issues like data isolation, scalability, flexibility and customization are all incorporated in the security consideration. Finally, the regulations that govern all tenants in a MTD must be made harmonised for adherence to the laws.

References

Abeysekera, S. (2001) Analysis approaches in participatory work involving ranks or scores. *DFID Theme Paper (revised).UK: Statistical Services Centre, University of Reading* [online].

Aulbach, S., Jacobs, D., Kemper, A. and Seibold, M. (2009) A comparison of flexible schemas for software as a service *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data.* [online]. Providence, Rhode Island, USA New York, NY, USA: ACM, pp.881-888.

Bezemer, C., Zaidman, A., Platzbeecker, B., Hurkmans, T. and Hart, A. (2010) Enabling multi-tenancy: An industrial experience report *Software Maintenance (ICSM), 2010 IEEE International Conference on.* [online]. pp.1-8.

Burns, R.B (2000) Introduction to research methods (4th Edition). SAGE Publications, London.

Chong, R., Carraro, G., Wolter, R. (2006) *Multi-tenant Data Architecture* [online]. [Accessed 23 August 2013]. Available at :<http://www.msdn.microsoft.com/en-us/library/aa479086.aspx#mlttntda_topic3>

Creswell, J. (2003) Research Design: Qualitative, Quantitative, and Mixed Methods Approaches (2nd Ed), Sage, Thousand Oaks, CA.

DeKimpe, D.M., Malloy, W.E. and Tomlyn, C.R. (2002) *Extension of data definition language (DDL) capabilities for relational databases for applications issuing DML and DDL statements* [http://www.google.com/patents/US6480848].

Elmore, A. J., Das, S., Agrawal, D. and Abbadi, A. (2011) Towards an elastic and autonomic multitenant database *Proc. of NetDB Workshop.* [online].

Fang, S. and Tong, Q. (2011) A comparison of multi-tenant data storage solutions for Software-as-a-Service *Computer Science and Education (ICCSE), 2011 6th International Conference on.* [online]. pp.95-98.

Gao, B., An, W., Sun, X., Wang, W., Fan, L., Guo, C et al. (2011) A Non-intrusive Multi-tenant Database Software for Large Scale SaaS Application *e-Business Engineering (ICEBE), 2011 IEEE 8th International Conference on.* [online]. pp.324-328.

Hui, M., Jiang, D., Li, G. and Zhou, Y. (2009) Supporting Database Applications as a Service *Data Engineering, 2009. ICDE '09. IEEE 25th International Conference on.* [online]. pp.832-843

Jacobs, D. and Aulbach, S. (2007) Ruminations on multi-tenant databases. *BTW Proceedings* [online]**,** **103**pp. 514-521 .

Keemti, P. (2010) *Multi-tenant Database Architecture* [online]. [Accessed 23 August 2013]. Available at <http://www.msdn.microsoft.com/eus/library/aa479086.aspx#mlttntda_topic1>

Mintzberg, H. (1973) The Nature of Managerial Work, Harper and Row, New York.

Ni, J., Li, G., Wang, L., Feng, J., Zhang, J. and Li, L. (2014) Adaptive Database Schema Design for Multi-Tenant Data Management. *Knowledge and Data Engineering, IEEE Transactions on* [online]**,** **26**(9), pp. 2079-2093 . .

Rea L. M. and Parker, P. A. (1997). *Designing and Conducting Survey Research*, 2nd Ed., Jossey-Bass Publishers, San Francisco, USA.

Schaffner, J. (2013), Multi tenancy for cloud-based in-memory column databases: workload management and data placement, Springer, Wien

Schiller, O., Schiller, B., Brodt, A. and Mitschang, B. (2011) Native support of multi-tenancy in RDBMS for software as a service *Proceedings of the 14th International Conference on Extending Database Technology.* [online]. Uppsala, Sweden New York, NY, USA: ACM, pp.117-128.

Vashistha, A and Ahmed, P. (2012)"SaaS Multi-Tenancy Isolation Testing- Challenges and Issues", *International Journal of Soft Computing and Engineering,*[online] **2**(5), pp. 49-50 pp. 49-50 Available at:<http://wlv.summon.serialssolutions.com>

Walraven, S., Van Landuyt, D., Truyen, E., Handekyn, K. and Joosen, W.(2014) Efficient customization of multi-tenant Software-as-a-Service applications with service lines. *Journal of Systems and Software* [online] (0), Available at:<http://www.sciencedirect.com/science>.

Wang, Z., Guo, C., Gao, B., Sun, W., Zhang, Z and An, W. (2008) A Study and Performance Evaluation of the Multi-Tenant Data Tier Design Patterns for Service Oriented Computing *e-Business Engineering, 2008. ICEBE '08. IEEE International Conference on.* [online]. pp.94-101.

Wood, K. and Anderson, M. (2011) Understanding the Complexity Surrounding Multitenancy in Cloud Computing *e-Business Engineering (ICEBE), 2011 IEEE 8th International Conference on.* [online]. pp.119-124.

Ying, H., Wang, Q., Wang, Z., and Wang, N. (2011) DB2MMT: A Massive Multi-tenant Database Platform for Cloud Computing *e-Business Engineering (ICEBE), 2011 IEEE 8th International Conference on.* [online]. pp.335-340.