**Journal homepage: www.sabauni.net/ojs**

**Saba Journal of Information Technology and Networking (SJITN)**

Journal homepage: www.sabauni.net/ojs

Saba Journal of Information Technology and Networking (SJITN)

**Journal homepage:** www.sabauni.net/ojs

**Saba Journal of Information Technology and Networking (SJITN)**

# Table of Contents

journal homepage: www.sabauni.net/ojs

Saba Journal of Information Technology and Networking (SJITN)

## Review

# Cloud computing advantages over Traditional e-learning

**Dr. Arwa Y. Aleryani*** [a], **Samah Alariki** [b]

[a] *Saba University* , [b] *Yemeni University*

**Article info**

**Article history:**

Accepted October, 2016

**Keywords:**

Cloud computing,
E-learning systems,
Security

**Abstract**

In this article, we displayed an overview of the current state of The Structure of Cloud Computing for e-learning. Our goal is to present how significant the idea of moving e-learning systems onto Cloud is? And what are the advantages of moving e-learning systems onto Cloud? It is well known that traditional e-learning was a huge offer from the developing technology to support learning systems, but it suffers from some problems; especially those related to budget issues. With the rapid growth of technology, the cloud computing is a new offer that helps overcome these problems and brings new environment for e-learning systems.

* Corresponding author: Dr. Arwa Y. Al-Eryani
E-mail: info@sjitn.sabauni.net

## 1.Introduction

Conventional classroom-based learning (traditional learning system) suffers from a lot of problems. It is expensive, takes a long time and the results can vary. It has limited attended location and time. Web-based learning offers several advantages over conventional classroom-based learning [1]. The e-learning approach offers an alternative that is faster, cheaper and potentially better.

E-learning is a computer based educational tool and system that enables any one to learn anywhere and at any time suitable for him/her. Among learning technologies, technology has advanced so much that the geographical gap is bridged with the use of tools that make students feel as if they were inside the classroom. At the same time, many disadvantages attacked e-Learning such as, they are weak on scalability at the infrastructure level. Several resources can only be deployed and assigned for specific tasks so that when receiving high workloads, the system needs to add and configure new resources of the same type, making resource achievement and management very expensive [2]. By time, with the technology's continuous development, cloud computing brought many advantages over the traditional e-Leaning.

## 2.Literature Review

Danny Manongga et al.[2]implemented a model which provides services such as, IaaS, PaaS, and SaaS, for e-learning in the educational environment in Salatiga. They used Moodle technology as e-learning applications that are installed on the Cloud. Moodle is a software package for a training purpose- web and internet based training commonly known as a Learning Management System (LMS), Course Management System (CMS), or Virtual Learning Environment (VLE). This study has shown the main components of the e-Learning system that is placed on OpenStack cloud computing. OpenStack is an open source cloud computing software for building dependable cloud infrastructures. OpenStack's aim is to allow any organization to build and offer cloud computing services using open source software running on standard hardware. The result of this study showed that the features of the cloud computing platform using the OpenStack method is quite appropriate for migration of learning system, so that it is able to form learning environments fully and efficiently, provide personalized contents, and facilitate the adaptation to the present model of education[2].

Abu El-Ala et. al, [3] presented in their research a creative environment derived from both virtual and personal learning environments based on cloud computing. This environment contains mixture of tools and techniques to improve the educational procedure. The proposed environment focuses on designing and monitoring educational environment based on reusing the existing web tools, techniques, and services to provide Browser-based-Application. The researchers tried to illustrate how to use cloud computing to improve the education process specifically in the Arab world. They stated that the problem is no longer to verify the importance of e-learning but it is to change to e-learning. Most educational institutions break down the barrier of establishment when they plan to move to e-learning systems which require many hardware and software resources. To solve such a problem the institutions can use the educational cloud. It provides a great solution to this problem, making it easy for any educational institution that wants to transform its system into an e-learning system. It will have two choices, whether to build its own private cloud or to go to a specific service provider to take a share in a pubic cloud after defining some parameters [3].

Fekry Fouad[4] highlighted in his paper the contribution of e-learning standards with the cloud computing standards and the impact on using cloud computing for e-learning systems. He provided an analysis for the important issues in current e-learning systems through a comprehensive comparison between e-learning systems before and after moving to a cloud computing environment. Fekry was using a generic frame work for cloud-based e-learning systems for his comparison. The results of his research showed that moving e-learning onto a cloud computing environment will support the e-learning in a large

deal. Cloud-based e-learning can reduce the cost of the development team, technical support team, testing effort, requirement elicitation load of daily backup management, and the cost of overall project expenditure [4].

## 3.Traditional E-learning

E-learning is the topic related to the virtualized distance learning by means of electronic communication mechanisms, specifically the internet. E-learning is the use of approaches with different functionalities (e-mails, Web pages, forums, learning platforms, and so on) as a support of the process of teaching/ learning. The e-learning is defined as an internet enabled learning. Components of e-learning can include contents of multiple formats, management of the learning experience, and an Online community of learners 'content developers and experts [1]'.

There are a lot of e-learning software that are used widely in various levels of education, such as universities, schools, and institutes. E-learning software provides a lot of benefits beyond classroom-based learning. The main benefit in the reduction of costs due to the IT physical environment is no longer needed [2].Also it can be accessed from any place, at any time. In addition, teachers can easily update the study materials that include multimedia contents in a user friendly approach, making them easier for students to understand the concepts. Finally, e-learning can be viewed as an approach that emphasizes the variation of learning material between teachers, reviewing their own materials for improvement at any time and from any place [2].

Among learning technologies, web-based learning offers several advantages over classroom-based learning. A physical environment is no longer required that leads to the reduction of costs, freedom of time and place. Moreover, the number of students is not limited by the area of a physical classroom. Additionally, the learning material is easy to be continuously updated. The teacher may also incorporate multimedia a content to provide a friendly framework and to facilitate the understanding of the concepts [5].

Some disadvantages can be listed for web-based learning (e-learning). First is the efficient use of these resources, where the lab PC and server will be underused in the evening and semester holidays. On the other hand, resource usage becomes very high during the day and the period of the semester. In addition it should also consider the costs associated with computer maintenance and licenses for every software package used. Second is the rapid growth of the number of students, teaching contents, services provided and resources available, has made e-learning system grow at an exponential level. With such problems, many educational institutions with limited resources and infrastructure face difficulty to continually keep up with this growth[2].

***E-Learning Security Concerns***

The basic security concern of e-learning technology usually arises when this type of learning is used to improve the functionality of traditional learning environment. These concerns are listed below [6]:

*1-User authorization and authentication:*

The user authorization is very essential and important when it comes to e-learning. In general the e-learners are from distant places, so a user ID and a password is essential. With the use of these two, one can login into the e-learning server and can access the features.

*2-Entry points:*

Entry points are the number of terminals where a possibility of security break may occur in the case of e-learning. As there are a number of clients in distant locations for each e-learning server, there are lots of entry points for each of them and the possibility of a security threat is more. The number of entry points should be reduced in order to get rid of this threat.

*3-Dynamic nature:*

One of the major concerns with the e-learning is that more processes are available in the dynamic sessions where a process can join and end the session without the notice of the others. This has opened much security infracts where they can easily attack the server and the client locations.

*4-Protection against manipulation:*

It is one of the key tasks to be implemented in an e-learning environment. It can be kept hidden from the other users by using certain techniques like digital signatures, firewalls etc. Similarly several other measures have to be taken in order to avoid manipulation from the registered users.

*5-Non-Repudiation:*

In the step of information security, cases of data loss or infection with virus, Trojan horse and other malicious treats are common. The system must be provided with the capability that the data is not damaged by these attacks.

***Social Aspects of Security***

In an online e-learning environment, students need to upload their soft copy of assignments. This kind of methods in e-learning technology brings the threats and vulnerabilities from internet to e-learning systems. To overcome these problems, basic security requirements such as the integrity, confidentiality and availability need to be applied. These security concerns are explained in more detail below [6]:

*-Confidentiality:*

Information and data sent Online should be kept confidential and not to be disclosed to an unauthorized third party. Under e-learning view, students like to get the assurance that their submitted soft copies of assignments Online, are kept discreet and only handed to their teachers on the e-learning environment.

*-Integrity:*

Information and data is not accidentally deleted or changed, and it should be kept accurate as in the original form. Students feel assured if integrity standards are maintained. This can only happen when their assignments submitted to teachers are kept safe in the original format without any further editing by others.

*-Availability:*

The reliable information should be present for access and modification by authorized people. Information present in e-learning servers must be present for students and teachers or other authorized people on a timely manner for their work. Students need assurance for uninterrupted reliable e-learning system to submit their assignments.

## 4.E-learning Based on Cloud Computing

With these disadvantages of e-learning based on internet, e-learning based on cloud computing is present to overcome these problems and provide a new environment for the e-learning system.

Cloud computing is the delivery of computing as a service rather than a product, whereby shared resources, software, and information are provided to computers and other devices and utilities [7]. It is a computing model in which IT resources provide a variety of services, and is available to users through the internet connection. Danny Manongga [2] defines the cloud computing as the provision of infrastructure, platform and software as a service (IaaS, PaaS, and SaaS) on the basis of simply, pay per use [2]. Although cloud computing is a technology services through the use of information on the internet, there are several requirements that must be met by an Internet-based service to be categorized as cloud computing [8], namely:

1) The services must be on demand.
2) The services must be flexible or scalable.
3) The services must be available and fully managed by the provider while users only need an Internet connection to operate the service.
4)The services must be measurable.
5) The services must be resource pooling. The cloud references a distributed collection of computing resources where the applications can be located anywhere on the accessible networks. In the cloud, there is a large pool of available virtualized resources such as hardware, development platforms, and ideally services [8], [2].

Cloud computing provides services at anytime, anywhere and that can be accessed from any device in such a way that the user is not responsible for where the services or applications are located, or how they are maintained or updated. All this and more will be the responsibility of cloud computing service providers [3].The five essential characteristics of a cloud model are: on demand self-service, broad network access, resource pooling, rapid flexibility, and measured

service [2].It has three service models; software, platform, and infrastructure. Cloud computing has four deployment models Private, Community, Hybrid, and Public Clouds [2];[6].

***Types of Clouds for Deployment***

There are different types of clouds for deployment [9]; [10]; [6]:

•*Public Cloud:*

A public cloud can be accessed by any subscriber with an internet connection and access to the cloud space.

•*Private Cloud:*

A private cloud is established for a specific group or organization and limits access to that group only.

•*Community Cloud:*

A community cloud is shared among two or more organizations that have similar cloud requirements.

•*Hybrid Cloud:*

A hybrid cloud is essentially a combination of at least two clouds, where the clouds included are a mixture of public, private, or community users.

***Cloud Computing Services***

Cloud computing services were listed as follows [4]; [11];[7]:

•*IaaS:*

Infrastructure layer corresponds to IaaS(Infrastructure as a service) is the lowest layer of the network. It is the supply of hardware as a service, that is, servers, net technology, storage or computation, as well as basic characteristics such as Operating Systems

•*PaaS:*

Platform layer corresponds to PaaS(Platform as a service) that made a higher level of abstraction on the base of Iaas layer. It is an integrated set of software with all the requirements that a developer needs to build applications, both for the developing and for the execution stages.

•*SaaS:*

SaaS(Software as a service) is a software distribution model, designed for web delivery, a user can deploy and access through the Internet hosting. It is to offer software as a service.

•*Cloud clients:*

Users who access cloud computing using networked client devices, such as desktop computers, laptops, tablets and smart phones.

Requirements for Building a Cloud Infrastructure

For the architect assigned with constructing a cloud infrastructure, there are seven key requirements that need to be addressed when building a cloud approach. These requirements include [12]:

*1-Heterogeneous Systems Support:*

cloud management providers must integrate traditional IT systems within in order to truly meet the requirements of the data center. Companies that don't support technologies from the likes of Cisco, Red Hat, NetApp, EMC, VMware and Microsoft will fall short in delivering a true cloud product that fits the needs of the data center.

*2-Service Management:*

Service offerings should include resource guarantees, metering rules, resource management and billing cycles. The service management functionality should tie into the broader offering repository such that defined services can be quickly and easily organized and managed by the end user.

*3-Dynamic Workload and Resource Management:*

In order for a cloud to be truly on-demand and elastic while consistently able to meet consumer service level agreements (SLAs), the cloud must be workload- and resource- aware. The system must be able to dynamically prioritize systems and resources on-the-fly based on business priorities of the various workloads to ensure that SLAs are met.

*4-Reliability, Availability and Security:*

To be fully reliable and available, the cloud needs to be able to continue to operate while data remains intact in the virtual data center, regardless if a failure occurs in one or more components.

*5-Integration with Data Center Management Tools:*

Within most data centers, a variety of tools are

used for provisioning, customer care, billing, system management, directory, security and much more. Cloud computing management solutions do not replace these tools and it is important that there are open application programming interfaces (APIs) that integrate into existing operation, administration, maintenance and provisioning systems (OAM&P) out of the box.

*6-Visibility and Reporting:*

The need to manage cloud services from a performance, service level, and reporting perspective becomes vital to the success of the deployment of the service. Data center operations have the requirement of having real-time visibility and reporting capabilities within the cloud environment to ensure compliance, security, billing and charge backs as well as other instruments, which require high levels of granular visibility and reporting.

*7-Administrator, Developer and End User Interfaces:*

One of the primary attributes and successes of existing cloud-based services on the market comes from the fact that self-service portals and deployment models shield the complexity of the cloud service from the end user.

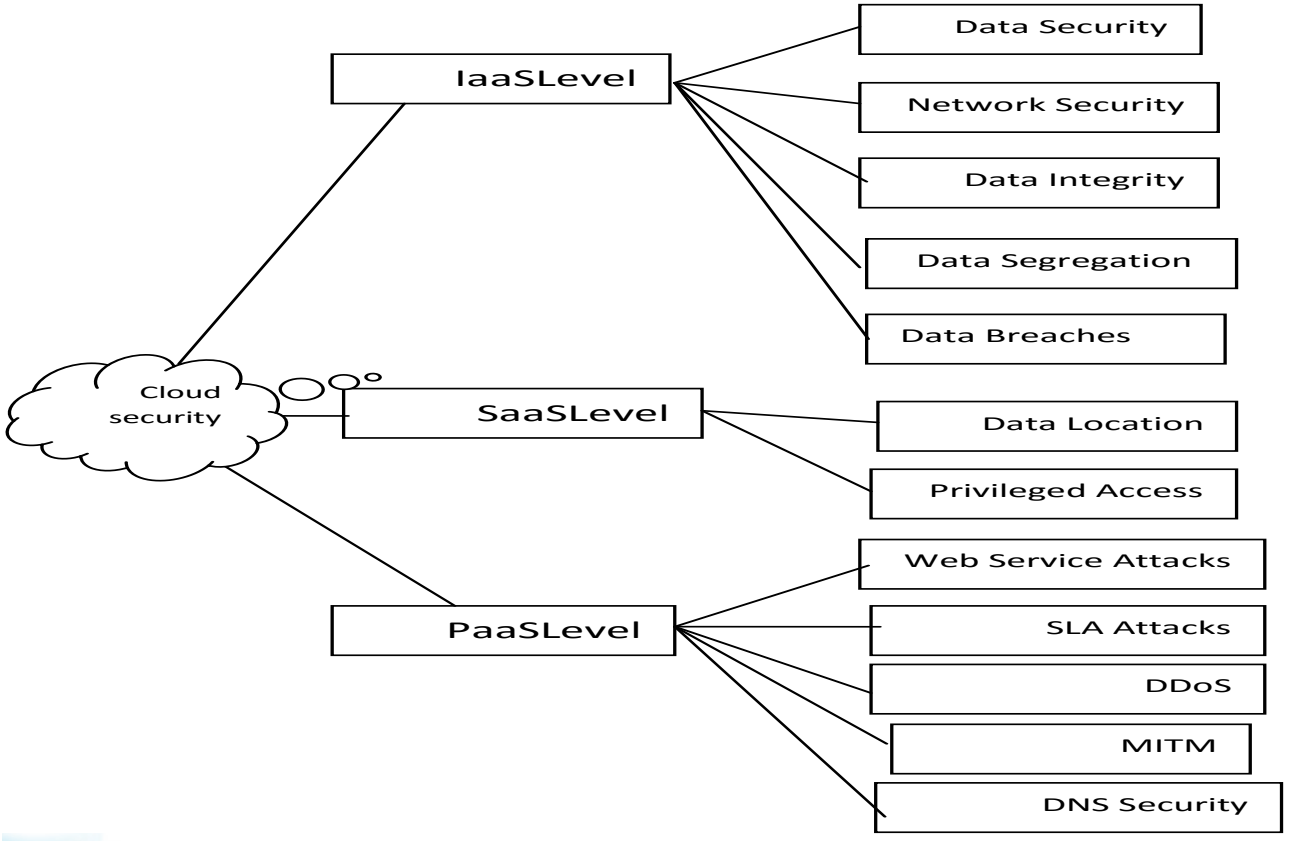
This helps by driving adoption and by decreasing operating costs as the majority of the management is offloaded to the end user.

***Cloud Computing Security***

Security is one of the primary concerns in the greater context of cloud computing as it relates to cloud based e-learning. From 2005-2011, security has been in the top four IT issues as published by Educause(a "non- profit association whose mission is to advance higher education by promoting the intelligent use of information technology")[13].When shifting e-learning into the cloud, the main security concerns are about confidentiality, integrity and availability. Security remains as an integral component of the top ten IT issues in 2012[13].

Security Attack Types

On the other hand there are many types of security attacks in cloud based e-learning as shown in figure (2): [14]




Figure(2 ): Types of security attacks [ Durairaj]

Cloud computing employs three service delivery models as listed below; through which different types of services are delivered to the end user. Each service model has different levels of security requirements in the cloud environment. These are Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS).

The types of security attacks for Software as a Service (SaaS) are:

*Data Security:*

Data plays a vital role in the cloud services, because many of cloud service providers store customers' data on large data centers. There is no guarantee for customers' data during transition operations. Data corruption may occur, when multiple devices are synchronized by one user. Data security can be classified into two ways: First, data owner must be content that the cloud service provider will only process the data according to the customer instructions. Second, data owner must be convinced that the cloud service provider has taken appropriate actions in cases like, during unauthorized data access, data modification, and destruction of data by intruders.

*Network Security:*

Enterprises store sensitive data in the cloud server and SaaS vendors can manipulate it. To protect data from leak-age of sensitive information, strong network traffic encryption techniques must be applied to manage data flow over the network, for example Secure Socket Layer (SSL) and Transport Layer Security (TLS).

*Data Integrity:*

Data integrity defines the correctness, accessibility, high quality, and reliability of stored data. Cloud provides integrity of data storages for customer privacy.

*Data Segregation:*

Data is located in the cloud in a shared environment (there are multiple tenants that are sharing a single location) so one customer's data is stored along with another customer's data. This can affect difficulty in data segregation. Customers should examine the cloud provider's architecture to make sure proper data segregation, and also customers should be aware of protocols and implementation methods of the encryption system.

*Data Breaches:*

Ever since data from different users and business concerns exist jointly in a cloud environment, breaking the data laws of cloud environment will certainly attack and damage the data of all the users. Therefore the cloud becomes a very important worthy target. However SaaS promotes declaration that SaaS providers can provide better security to customers' data than by conservative means. Still insiders have rights to use the data in a different way.

The types of security attacks for Platform as a Service (PaaS) level are:

*Data Location:*

PaaS vendors provide services for application design, application development, deployment, team collaboration, web service integration, and testing. In this statement, the PaaS cloud users access the applications of SaaS providers to get services so that the customer does not know where the data is stored and processed.

*Privileged Access:*

The cloud provider has the full right to access data including other users of the cloud and other third party suppliers, once data is stored in the cloud environment. Two approaches could be used by the data owner to maintain the privileged user access. First, is to choose a strong encryption method for storing data and using another encryption method for data access; second, is to maintain a high standard of confidentiality of data, legally imposing the requirements of the cloud provider through contractual responsibilities and assurance mechanisms.

The types of security attacks for Infrastructure as a Service (IaaS) level are:

*Web Service Attacks:*

Web service protocols are used by cloud users for getting services. Simple Object Access Protocol(SOAP) is the most balanced protocol in web services. A standard extension of security in SOAP is web service security, addresses the security of web services.

*SLA Attack:*

When customers have transferred their core business functions onto their committed cloud environment, they should be ensured of the quality, availability, reliability, and performance of these resources, because cloud users do not have control over these computing resources. Cloud users are expected to get guarantees from cloud providers on service delivery, which is rendered through Service Level Agreements (SLAs) to manage among cloud providers and cloud users.

*DDoS Attack:*

Distributed Denial of Service (DDoS) attack is an advanced version of denial of services in terms of denying the important services by giving large number of requests, which is not handled by target server.

*MITM Attack:*

Man In The Middle (MITM) attack is encountered when an attacker directs himself between two legitimate users. This attack is also a class of eavesdropping. The attacker sets up the connection between two users and tries to hear the communication or it reveals false information between them. To avoid these kinds of attacks, tools have been developed like, Dsniff, Cain, Ettercap, Wsniff, Airjack etc.

*DNS Attack:*

IaaS cloud environment deals with a risky attack vector known as DNS Attack, which translates the domain name to an IP address. The user using IP address is not realistic because it has been routed to some other cloud virtual machine instead of original address expected.

***Comparison Between E-learning Before and After Moving to Cloud Computing***

According to Fekry Fouad [4]the comparison between e-learning before and after moving to cloud computing can be determined as follows: Before moving to cloud computing, many requirements are needed; such as monitoring of client logs and information by a third party, need for technical IT support for failovers, need for an e-learning system development team, need for extra hardware and software resources, need to configure latest technology updates, need to arrange extra power and cooling, need for requirement gathering and elicitation , need for project management, need for coding, need for testing, need for deployments, need for daily storage and backup, all required finely huge costs. On the other hand, after moving to cloud computing, there was no need for all these requirements.

Many problems that face e-Learning before moving to cloud computing could be solved by cloud computing. Some of these problems are the conflict between opposing goals of different clients, either play it together, if not, they need to separate, higher risks of resource availability and failure, lack of trust in data modification before storing, rejection of service attacks in critical server health, higher risks of stress, load and congestion, difficulty to review, lack of computation and accuracy trust, lack of confidentiality, lack of trust on security policies and access control [4].

***Advantages of E-learning Based on Cloud Computing Over Traditional E-learning***

There are consequences and implications associated with the development of e-learning in the cloud environment as confirmed by Masud et.al,[15] : Accessed via the Web, does not require any software on the client-side, pay per use, SaaS Server can support many educational institutions, all customer data is at SaaS Server[15]. Ouf et.al, [16] give several values of cloud computing potential for education including :

1. No back up required to drive and transfer from one platform to another
2. No crash recovery needed,
3. Access from a variety of places, and
4. Flexibility. Furthermore, no stolen data, virtualization, centralized data storage, monitoring data access becomes easier[16]. Cloud computing not only saves the money needed for upgrading many labs' hardware or purchase many software licenses but it also reduces periodic maintenance operations. It also provides a high level of security and privacy [3]. Also, cloud-based e-learning can reduce development team cost, technical support team cost, testing effort, requirement elicitation, load of daily backup management, and cost of overall project expenditure [4].

Madan[11]mentioned some of cloud computing advantages like reduced upfront investment (such as software, hardware, and professional staff to maintain servers and upgrade software), reduced launching time (where days become hours) expected performance, high availability, infinite scalability, great fault tolerance capability (fault tolerance is the property that enables a system to continue operating properly in the event of the failure of some of its components), and improved collaboration, accessibility, and also mobility, allowing users to use any device(such as a personal computer (PC), or a mobile phone).

Also Riahi [17] listed some of the cloud computing advantages which can be summarized as low cost, improved performance, instant software update, improve the compatibility of document formats, benefits for students, benefits for teachers, and information security [17].

Alghali[18] listed many advantages to educational institutions when using cloud computing as follows [18]: Support from the service provider can be gained. The availability where in cloud computing, systems can automatically detect the node failure and exclude it without affecting the users system. Also, to reduce costs as cloud computing gives an opportunity to reduce the amount of money spent on IT. As well, to offer the service on-demand where cloud computing users have access to computing capabilities with 24 hours access to the infrastructure, content and software when needed without requiring human interaction with the cloud service provider CSP. Add to that, pay per use where the cloud computer users pay only for what they actually use. Another advantage is Wide Network Access, the education institutions can enhance their communication by participating with each other. Another one is Rapid Elasticity, services in cloud computing have the ability to scale resources both up and down on demand.

While the cloud based e-learning has many benefits for educational institutions, still, there are some challenges in implementing cloud computing to e-learning technology. These challenges in cloud based e-learning technology are listed below. Alghali [19] listed some of them as privacy where data privacy protection in cloud computing faces many challenges such as access, storage, compliance, retention, destruction, audit and monitoring, breaches and controversy of various legal systems. Security also plays a necessary condition of some e-learning materials. If the data is stored in cloud, the question of the security of this valuable data on unknown cloud servers arises. Reliability is a great concern as well for education institutions Legal Issues. If an organization wants to take the advantage of cloud computing system services, it has to make sure which countries are hosting their private data, and what are the country's laws that govern data [20].

## 5.Conclusion

We can conclude from the previous studies that the cloud computing is the best solution for all educational institutes and universities. It is not a question of reducing costs or infrastructure, rather a more efficient work and secure environments. Nowadays, all institutes and universities especially those with limited budgets have to benefit from cloud computing and improve their e-learning systems by gaining all the advantages of cloud computing. Ongoing technology development will continue moving, and will always offer something new to the world; this is the charge that has to be paid to keep the organizations up to date with dynamic technological development. All organizations have to keep reducing the gap between their current situation and the new development, in order to continue offering their services in a sufficient way. The researchers believe that cloud computing is the next big trend for an efficient e-learning system. This is for the features and capability to improve traditional e-learning system both technologically and cost wise.

## References

[1] Fernández, A. ; Peralta, D. ; Benítez, J. ; Herrera, F. ( 2014 ) "E-learning and educational data mining in cloud computing: an overview " , Int. J. Learning Technology, Vol. 9, No. 1, 2014.

[2] Manongga, D. ; Herry, W. ; Hendry (2014) "E-Learning Development as Public Infrastructure of Cloud Computing" , Journal of Theoretical and Applied Information Technology 10th April. Vol. 62 No.1

[3] Abu El-Ala, N. S. ; Awad, W. A. ; El-Bakry, H. M (2012) "Cloud Computing for Solving E-Learning Problems" International Journal of Advanced Computer Science and Applications(IJACSA), Vol. 3, No. 12, 2012.

[4] Fekry Fouad Ahmed (2015) "Comparative Analysis for Cloud Based e-learning" International Conference on Communications, management, and Information technology (ICCMIT'2015), Volume 65, 2015, Pages 368-376 , ELSEVIRE

[5] Jolliffe, A., Ritter, J. and Stevens, D. (2001) "The Online Learning Handbook: Developing and Using Web-based Learning", Kogan Page, London.

[6] Kumar, G.; Chelikani, A. (2011) "Analysis of Security Issues in Cloud based E-Learning", Master's thesis in Informatics , 2011MAGI23.

[7] Fernandez A.; Peralta,D. ; Herrera, F.; Benıtez J. (2012) "An Overview of E-Learning in Cloud Computing" L. Uden et al. (Eds.):Workshop on LTEC 2012, AISC 173, pp. 35–46. springerlink.com _c Springer-Verlag Berlin Heidelberg 2012

[8] Chang, W.Y., Abu-Amara, H., and Sanford, J.F., (2010) " Transforming Enterprise Cloud Services", Springer Dordrecht Heidelberg London New York

[9] Huth, A. ; Cebul, J. (2011) "The Basics of Cloud Computing",Carnegie Mellon University, Produced for US-CERT, a government organization

[10] Abu-Shanab, E. ; Qasem H. (2014 ) "Cloud computing Adoption: Brand Equity impact on Users' choice" , Saba Journal of Information Technology and Networking vol.2 no. 1

[11] Madan, D.; Pant, A.; Kumar, S.; Arora, A. (2012 ) "E-learning based on Cloud Computing" , International Journal of Advanced Research in Computer Science and Software Engineering Volume 2, Issue 2, February 2012 ISSN: 2277 128X

[12] http://www.cio.com/category/cloud-computing/

[13] Hameetha, S, ; Sheeba, T. ; Nisha, S. ( 2013 ) "Security in Cloud based E-Learning" , International Journal of Advanced Research in Computer Science and Software Engineering Volume 3, Issue 1, January 2013 ISSN: 2277 128X

[14] Durairaj, M.; Manimaran, A.( 2015) "A Study on Security Issues in Cloud based E-Learning", Indian Journal of Science and Technology, Vol 8(8), 757–765, April 2015

[15] Masud, A.H., Huang, X., (2011) "ESaaS: A New Education Software Model in E-learning Systems". In: Zhu, M. (ed.) ICCIC 2011, Part V. CCIS, vol. 235, pp. 468–475. Springer, Heidelberg

[16] Ouf, S., Nasr, M., (2011) " Business intelligence in the cloud". In: IEEE 3rd International Conference on Communication Software and Networks (ICCSN 2011)

[17] Riahi, G. (2015) "e-learning systems based on cloud computing: a review" , the 2015 international conference of soft computing and software engineering (SCSE 2015), ELSEVIRE

[18] Alghali, M.; Najwa, H. ; Roesnita, I. ( 2014) "Challenges and Benefits of Implementing Cloud Based E-Learning in Developing Countries" Faculty of Science and Technology UniversitiSains Islam Malaysia [USIM] , Nilai, Malaysia

[19] Alghali, M.; Alwi, N. H. M., ; Ismail, R. (2013) "Privacy in Cloud Based E-Learning". In The Second International Conference on Informatics Engineering & Information Science (ICIEIS2013) (pp. 355-362). The Society of Digital Information and Wireless Communication.

[20] Zhou, M., Zhang, R., Xie, W., Qian, W., & Zhou, A. (2010). Security and privacy in cloud computing: A survey. In Semantics Knowledge and Grid (SKG)", 2010 Sixth International Conference on (pp. 105-112). IEEE.

journal homepage: www.sabauni.net/ojs

Saba Journal of Information Technology and Networking (SJITN)

# Article

## The formalization information model of scheduling university using language Tutorial D

**Dr. Amr Abu Hania**

*Platov South Russian State Polytechnic University*

**Article info**

**Article history:**

Accepted Nov, 2016

**Keywords:**

Relational language ,
Object-Oriented Database,
Process ,
Timeslot and schedule.

**Abstract**

The University Class Scheduling Problem is concerned with assigning a number of courses to classrooms, taking into consideration different constraints like classroom capacities and university regulations. It also attempts to optimize the performance criteria and distribute the courses fairly to classrooms depending on the ratio of classroom capacities to course enrollments. The problem is a classical scheduling problem, and is considered to be NP-complete. Several formulations of information models have been proposed to solve scheduling problems, most of which are based on local search techniques. In this paper, we propose a complete approach using two relational database languages that are associated with this model and that are the Structured Query Language (SQL) and Tutorial D. This research describes the composition of information tables, as well as their structure and relationships. The relational model of data has a strong influence on the database field and cannot be discarded for any future direction regarding the development of databases.

* Corresponding author: Dr. Amr Abu Hania
E-mail: am27er@gmail.com

## 1. Introduction

Since the introduction of the relational model of data [7], relational database management systems came into existence and have proven to be dominant in the database field. These are software products that can be used in creating and managing-relational databases. These software products are all based on relational database languages that give clients the ability to interact with data that is stored in relations. There are many relational database languages in the market but the most popular and standardized one is the Structured Query Language (SQL).

The purpose of this paper is the description of the structure of the model scheduling problems in the language Tutorial D. Database programming language called Tutorial D was defined to replace SQL. Tutorial D is to form a solid foundation in which the Third Manifesto can be built upon. This foundation must be originating from the roots of the relational model of data that was proposed by [7] without any existence of SQL. This is due to the many deficiencies that provide a detail explanation of those SQL deficiencies.

## 2. Tutorial D

Tutorial D is a relational language with the purpose of forming a solid foundation for future database management systems. It all started when Date and Darwen observed some of the trends which attempted to integrate object and relational technologies but in an ill-defined manner. The first of these attempts was the Object-Oriented Database System Manifesto which was proposed by [8]. It considered object-oriented database systems to be theoretically and experimentally useful, and thus, needed a common data model to formalize those systems.

According to Atkinson et al. (1989), the reason behind this is that object-oriented database systems lacked a common data model before the introduction of the Object-Oriented Database System Manifesto. However, Date and Darwen (2000) criticized this approach to be ignoring the relational model of data,and thus, not suitable for a future direction of database management systems to be moved along. This is due to the importance of the relational model of data, as described earlier. The Second attempt came along after the Object-Oriented Database System Manifesto and was called the Third-Generation Database System Manifesto. Date and Darwen[4,5] proposed a third attempt of directing the next generation of database management systems. A manifesto was formed and was called The Third Manifesto, with the purpose of forming a theoretically correct mechanism of building object/ relational database management systems which evolve from the original relational model of data but not replace it, because of its importance and relevance to the database field [4].Subsequently, a new database programming language called Tutorial D was defined to replace SQL[6].

The purpose of Tutorial D is to form a solid foundation in which the Third Manifesto can be built upon. This foundation must be originating from the roots of the relational model of data that was proposed by [7] without any existence of SQL[3,4,5].

### *2.1 Research Problem*

In the base of the mathematical model of the subject area “Schedule”, the concept of “process” is considered the most significant ( a set of interrelated resources and activities that are linked to a specific time interval). Let’s perform the formalization of the concepts “resource”, “time interval”, “activity” and “schedule” within the subject area.

The resources in this case are courses, lecturers, classrooms, and time slots. For example, one obvious constraint is that a lecturer cannot teach more than one course at the same time. In the base of mathematical model of the subject area”Schedule” the concept of “process” is considered the most significant. It is a set of inter-related sources and activities that are linked to a specific time interval. Let’s perform the formalization of the concepts”resource”, “time interval”, “activity” and “schedule”within the subject area [1,2].

There are many deficiencies and flaws surrounding the use of SQL as a relational database language. In an attempt to avoid using SQL, Tutori-

al D was introduced as anew relational database language that addresses SQL deficiencies and introduces new feature which realizes the full potential of the relational model of data[4]. However, due to the popularity of SQL, adopting Tutorial D can be difficult. Hence, the need to find a suitable mechanism to address these issues is crucial. One suggested solution is to form database in Tutorial D in order to allow legacy SQL clients to inter operate with Tutorial D databases[3,5,6]. As a resource we consider the university classroom assets where it is a multi-index.

### *2.1.1 The Formalization of the Essence of "Resource"*

Let us describe using the basic structures of the language relational relations associated with the formalization of the concept of "resource" [1,5]: (buildings for classrooms - BCR), (types of classrooms for use - TCRU), (types of classrooms on capacity - TCRC) and (classrooms) - CR:

```
VAR BCR BASE RELATION
{NB# INTEGER,
 NAME CHAR}
 PRIMARY KEY {NB#};
VAR TCRU BASE RELATION
{TCRU# INTEGER,
 NAME CHAR}
 PRIMARY KEY {TCRU#};
VAR TCRC BASE RELATION
{TCRC# INTEGER,
 MAX_CAP INTEGER,
 NAME CHAR}
PRIMARY KEY {TCRC#};
VAR CR BASE RELATION
 {NCR# INTEGER,
 NB# INTEGER,
 TCRU# INTEGER,
 TCRC# INTEGER,
NAME CHAR,
CAPACITY INTEGER}
PRIMARY KEY {NCR#},
FOREIGN KEY {NB#}
REFERENCES BCR,
FOREIGN KEY {TCRU#}
REFERENCES TCRU,
FOREIGN KEY {TCRC#}
REFERENCES TCRC;
```

Attributes of these relations have the following meanings: NB #, TCRU #, TCRC #, NCR # - unique keys corresponding tables; NAME - the name of ((buildings for classrooms , types of classrooms for use - TCRU, types of classrooms on capacity and classrooms); MAX_CAP - capacity audience of this kind; CAPACITY - capacity of a specific audience.

### *2.1.2 The Formalization Entity "Time Interval"*

When formalizing the essence timeslot (time interval) we note, first of all, that timetable is compiled for a specified period of training. Let us now describe the process of formalizing the notion of a time interval in the language of Tutorial D[3,4,5]. Basic relational relationship: (training weeks - TW), (days of the week - DW), (timeslots in the day - TD), (timeslots in the period - TP):

```
VAR TW BASE RELATION {
NW# INTEGER,
NAME CHAR}
 PRIMARY KEY {NW#};
VAR DW BASE RELATION
{ND# INTEGER,
 NAME CHAR}
 PRIMARY KEY {ND#};
VAR TD BASE RELATION
{NTD# INTEGER,
TIME INTEGER}
PRIMARY KEY {NTD#};
VAR TP BASE RELATION
{NT# INTEGER,
NW# INTEGER,
ND# INTEGER,
NTD# INTEGER}
PRIMARY KEY {NT#},
FOREIGN KEY {NW#}
REFERENCES TW,
FOREIGN KEY {ND#}
REFERENCES DW,
 FOREIGN KEY {NTD#}
REFERENCES TD;
```

### *2.1.3 The Formalization Entity of "Activity"*

Begin the process of formalizing the notion of activity in the language Tutorial D to determine the following relational relations: (course title - CT),

(types of lessons - TL), (courses - CRS), (departments of university - DU), (positions - PST), (teachers - TCH):

```
VAR CT BASE RELATION
{NCT# INTEGER,
 NAME CHAR}
PRIMARY KEY {NCT#};
VAR TL BASE RELATION
{NTL# INTEGER,
 NAME CHAR}
 PRIMARY KEY {NTL#};
VAR CRS BASE RELATION
{NCRS# INTEGER,
 NCT# INTEGER,
 NTL# INTEGER}
PRIMARY KEY {NCRS#},
FOREIGN KEY {NCT#}
REFERENCES CT,
FOREIGN KEY {NTL#}
REFERENCES TL;
VAR DU BASE RELATION
{NDU# INTEGER,
 NDU_P# INTEGER,
 NAME CHAR}
PRIMARY KEY {NDU#},
FOREIGN KEY {NDU_P #}
REFERENCES DU;
VAR PST BASE RELATION
{NPST# INTEGER,
 NAME CHAR}
 PRIMARY KEY {NPST#};
VAR TCH BASE RELATION
{NTCH# INTEGER,
 NPST# INTEGER,
 NDU# INTEGER,
 L_NAME CHAR,
 F_NAME CHAR,
 S_NAME CHAR,
 SHORT CHAR}
PRIMARY KEY {NTCH#},
 FOREIGN KEY {NPST#}
REFERENCES PST,
FOREIGN KEY {NDU#}
REFERENCES DU;
```

### *2.1.4 The Formalization of Entity "Learning Groups"*

Note that between the entities "learn" and "group", there is the relation "many to many".Relational attitude (study groups) can be described as follows[1,2,3]:

```
VAR GROUPS BASE RELATION
{GROUP# INTEGER,
 NAME CHAR,
NDU# INTEGER,
NUMBER# INTEGER}
PRIMARY KEY {GROUP#}
FOREIGN KEY {NDU#}
REFERENCES DU;
```

In this respect, it is assumed that each study group GROUP# belongs to only one department of the university NDU # (department or faculty), although in practice there are cases of "mixed" flows, including academic groups from several faculties and so the attitude GROUPS still require clarification. In addition, each group has a NAME, for example "FITU-1-5, 5B, FMF 1-2, 3" and for each group the number of students in it NUMBER #.

### *2.1.5 The Formalization of Entity "Learning Unit"*

To complete the formalization of the concept of operations, we make a record in the language Tutorial D (teaching unit - U)

```
VAR U BASE RELATION
 {NU# INTEGER,
 NG# INTEGER,
NCRS# INTEGER,
 NTCH# INTEGER}
PRIMARY KEY {NU#}
FOREIGN KEY {NG#}
REFERENCES GROUPS,
FOREIGN KEY {NCRS#}
REFERENCES CRS,
FOREIGN KEY {NTCH#}
REFERENCES TCH;
```

### *2.1.6 The Formalization of Entity "Schedule"*

The above mentioned formalization of such concepts as a resource, is a temporal interval and an activity that allows examining a time-table (schedule) as some unambiguous reflection of from setsU . In a set (work is Cartesian)

$T \times A$ :

$$s: U \rightarrow T \times A \in S$$

When choosing a format for the schedule in the computer's memory, it is convenient to introduce the set C={c1,c2,…,cm}={(t1,a1),…,(t1,anA),…,(tnT,a1),…,(tnT,anA)}, which we call the " Schedule Planning Grid "[1,2,5]. This - Table- empty graph, each cell corresponds to a pair "timeslot-audience". Scheduling in these cells is placed ("are assigned") in training units.

| | $(t_1,a_1)$ | $(t_1,a_2)$ | ... | $(t_1,a_{nA})$ | $(t_2,a_1)$ | ... | $(t_{nT},a_{nA})$ |
|---|---|---|---|---|---|---|---|
| $u_1$ | 0 | 1 | ... | 0 | 0 | ... | 0 |
| $u_2$ | 0 | 0 | ... | 0 | 1 | ... | 0 |
| ... | ... | ... | ... | ... | ... | ... | ... |
| $u_{nU}$ | 0 | 0 | ... | 1 | 0 | ... | 0 |

We shall assume that each grid cell has a unique number (an integer in the range 1 ... m),by which it can uniquely identify and get all necessary supporting information: Code classroom and code timeslot, where you can find the week number, day of the week and the number of training "pair".Example illustrates the process of learning the accommodation unit in the schedule planning grid as shown in Figure 1.

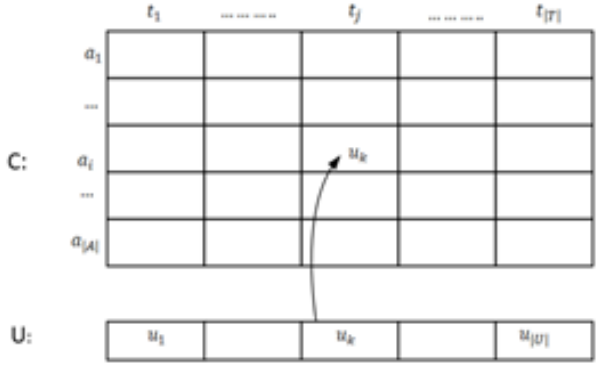



*Fig.1. Placing the training unit in the Schedule Planning Grid*

Schedule Planning Grid – SPG, can be represented by the following relational relationship:

VAR SPG BASE RELATION

{NCELL# INTEGER,
NCR# INTEGER,
NT# INTEGER}
PRIMARY KEY {NCELL#}
FOREIGN KEY {NCR#}
REFERENCES CR,
FOREIGN KEY {NT#}
REFERENCES TP;

NCELL# - unique key record (code grid cell), NCR# - code audiences, NT# - at timeSlots
At the end of the process of formalization of the basic objects (entities) of the problem, we present a referential attitude to the schedule:

VAR S BASE RELATION

{NS# INTEGER,
NU# INTEGER,
NCELL# INTEGER}
PRIMARY KEY {NS#}
FOREIGN KEY {NU#}
REFERENCES U,
FOREIGN KEY {NCELL#}
REFERENCES SPG;

There NS# - code element schedules, NU# - code teaching unit, NCELL# - code Schedule Planning Grid[1,2].

## 3.Conclusion and Future Work

In this research a new methodology for solving university course scheduling is proposed and implemented using real data set from the South Russia State Technical University. To Conclude this research, we can say that it was found that Tutorial D addresses the SQL deficiencies. This is achieved through providing true relational features that include rejecting duplicate tuples, NULLs and attribute ordering. Hence, utilizing Tutorial D can be useful in order to conform to the relational model of data and avoid SQL deficiencies. The advantages of this approach are:
-Synergistic effect of the combination of severity of each of notations.
- Problem solving followed by computer implementation models.
-Tasks, with the use of database technology, oriented to the client-server architecture.
-Using data structures in memory, which is typical for desktop applications or "thick" clients.

## References

[1] Ivanchenko A.N., Abu hania A.(2013),"information model scheduling problems for the university" News . institutions of higher learning. North -Caucasian region.Tech. Sciences -.№ 3 (172).C. 14-16.

[2] Ivanchenko A.N., Abu hania A. (2012) "On th eformulation scheduling problems for the university" News . institutions of higher learning. North -Caucasian region.Tech. Sciences -.№ 6. -C.9-13.

[3] Date, C. J. (2013). Relational Theory for Computer Professionals. " O'Reilly Media, Inc.".

[4] Date, C. J., & Darwen, H. (2007). Databases, types and the relational model: the third manifesto. Addison-Wesley Longman.

[5] Date, C. J., & Darwen, H. (2010). Database Explorations: Essays on the Third Manifesto and Related Topics. Trafford publishing.

[6] Almulla, A. Y. (2007). Translating between SQL and Tutorial D.

[7] Codd, E. F. (1970). A relational model of data for large shared data banks. Communications of the ACM, 13(6), 377-387.

[8] Atkinson, M. P., Bancilhon, F., DeWitt, D. J., Dittrich, K. R., Maier, D., & Zdonik, S. B. (1989, December). The Object-Oriented Database System Manifesto. In DOOD (Vol. 89, pp. 40-57).

**Journal homepage:** www.sabauni.net/ojs

**Saba Journal of Information Technology and Networking (SJITN)**

# Article

## Using XML as NoSQL Database in .NET and Comparing it with SQL Database

**Dr.Abdualmajed Al-Khulaidi**

*Sana'a University, Yemen*

**Article info**

**Article history:**

Accepted Jan, 2016

**Keywords:**

NoSQL Database,
SQL Database,
XML,
HTML .

**Abstract**

This scientific paper intends to explore the use of XML as a database as being a NoSQL type and the functions it carries out. Some of its important functions are to store and transfer data. It also tackles another point as well; how to use XML database in .NET by using C# language. To clarify such points, examples are provided to show the processes of adding, reading, deleting, updating and searching for data by using XML in C#. The paper further draws a comparison between NoSQL and SQL databases. The comparison is conducted to incorporate a number of points: database structuring, type of data they store, querying, scaling, support, reliability, and need for storing complex data and querying for it.

* Corresponding author: Dr. Abdualmajed Al-Khulaidi
E-mail: alkhulaidi@mail.ru

## 1. Introduction

Before initiating this paper, it is important to introduce the types of databases. There are several types among others are: network database, hierarchical database, relational database, object-oriented database …etc. Most famous of which is the relational database as its fame is similar to that of Oracle, SQL Server, MySQL, Access. More importantly is that relational databases store or organize data in tables, which are linked in the form of relational models. Another set of database, OOP Database, relies on having objects in their structure [1,2,3,4,5,6]. One more type is Document Store or NoSQL database in which data is stored in the form of encoded files in XML or JSON format or in the form of Word or PDF format.

XML is not a programming language, yet it is designed to transmit and store data. It is a family of Extensible Markup Language which also includes the well-known HTML language. This language is a subset of the Standard Generalized Markup Language SGML which first appeared in 1960, that is, about 30 years before the World Wide Web came to function. It assists in marking up and coordinating as well as organizing exchanged documents and emails via the Net. The family of marking up languages is featured by using tags in the form of parenthesizes such as < >, and having a tree-like structure. While it lacks any instructions or processes, its function is limited to mark up contents with certain codes that are understood by browsing software. It is known that HTML is a descriptive language used to display data on a web page. While some think that XML language is different from HTML, it is possible to argue that what links both languages is that they are subsets of the same language. It is well known that HTML uses a very limited number of tags which enables browsers to interpret these tags or codes automatically in order to coordinate the proper display of data on a web page.

XML files do often offer benefits in terms of not needing mysql databases or other similar databases. The benefit is incorporated in making a file writable and enabling it to store information that is transmitted to it. Via using XML files, it is possible to save and display data in an HTML page without the need for any programming languages. An example of XML which allows the saving of information is a CSS file permitting the saving of styles.

### *1.1 XML Tree Structure*

XML documents are formed as element trees.
An XML tree starts at a root element and branches from the root to child elements [7,8].
All elements can have sub elements (child elements):

```
<root>
  <child>
    <subchild>.....</subchild>
  </child>
</root>
```

### *1.2 An Example XML Document*

```
<?xml version=”1.0” encoding=”UTF-8”?>
<bookstore>
  <book category=”cooking”>
    <title lang=”en”>Everyday Italian</title>
    <author>Giada De Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>
  <book category=”children”>
    <title lang=”en”>Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
  <book category=”web”>
    <title lang=”en”>Learning XML</title>
    <author>Erik T. Ray</author>
    <year>2003</year>
    <price>39.95</price>
  </book>
</bookstore>
```

### *1.3 XML Functions*

XML functions could be classified into three types [9,10]:
A benefit could be obtained from XML within one website:

1. *This benefit is often two faced.*

The use of XML files as databases:
This could be through either through the use of

XML documents to store data or the conversion of a group of documents into a database by using one of the following applications.

## 2. Making use of XML files when exchanging data among two or more websites

We often hear and realize the great cooperation in information and data sharing among various websites. Such a feature allows us to easily post our favored links in Delicious on our Facebook and Twitter pages and other social networking websites. The users of Good reads website also display their updates on Twitter and Facebook. A sufficient example for data/ information sharing among websites is Friend Feed which displays summaries of most social networks. All of that is being done through XML, that is, either through RSS files or other XML files.

## 3. Making use of XML when sharing data on computers or other devices and various applications

Let's take Twitter as an example. I can log in to my Twitter account and read my friends' updates on Twitter through:
www.twitter.com, my mobile phone, Desk/Laptop Computer. All of these means deal with the same database, and the same updates are displayed. Yet, each means/ device is programmed by a different language and functions differently. All of this is done by the benefits and features offered by XML.

2. Using XML Files as NoSQL Databases in .net

NoSQL is a new model or type of database management system. It follows a different model than the older and traditional models which follow the model of linked or tied tables ( relational databases). The most outstanding difference between these two models is the use of tables. Unlike relational database, NoSQL does not consider tables as the foundation element for constructing database. For such a reason, NoSQL is used as an alternative or substitute for SQL language in terms of dealing with data.

XML is one model of a NoSQL database and it is not a programming language. Yet, we could state that it takes the form of textual writings, composed with known and specific rules. It is very similar to HTML in terms of writing its codes by using tags. Unlike HTML, XML is not limited in terms of the use of words; a designer could use any word to create the root and nodes and sub-nodes. In addition, XML is used to store data regardless of how they are displayed. An XML file is of a small size compared to database files and is easily accessed and dealt with. Some of its functions may include the storing and sharing of data among databases. A table could be exported to an XML file while an XML file could be imported. In addition, an XML file is composed of a root, nodes and sub-nodes. The attribute of a node could be specified, a matter which is known to be "Attribute." Such a linguistic item represents the characteristic of a node or (an element). In .net (dot net), XML is dealt with through LINQ. LINQ is from Microsoft to unite the means of using data irrespective of data sources. The Net Framework library has offered a means to easily deal with such files through XML or XML. LINQ libraries. These two libraries provide objects and functions/subroutines enabling us to deal with XML through creating a file, or updating the file or even deleting or performing search in it.

Let's take a look at the following example, illustrating how to use XML in C#.

***Required Libraries:***

```
using System;
using System.Collections.Generic;
using System.Drawing;
using System.Globalization;
using System.Linq;
using System.Windows.Forms;
using System.IO;
using System.Xml;
using System.Xml.Linq;
```

We design a Class by the name of Emp as follows:

```
class Emp
    {
        public int ID { get; set; }
        public string Name { get; set; }
        public string Gender { get; set; }
        public int Age { get; set; }
        public string Imgstring { get; set; }
```

```
}
```

***General Variables:***

```
private XElement _xel,_empinfo, _id, _name, _age,_image;
private readonly XmlDocument _doc = new XmlDocument();
 private XmlNode _node;
private readonly List<Emp>  _emp = new List<Emp>();
 private int _pos;
 private const string Xpath = @"D:\emp.xml";
  private string _imgstr = "";
```

Function Filling Field

```
private  void Fill(int pos)
{
id_txt.Text = _emp[pos].ID.ToString(CultureInfo.InvariantCulture);
 name_txt.Text = _emp[pos].Name;
age_txt.Text = _emp[pos].Age.ToString(CultureInfo.InvariantCulture);
pic.Image = StringToImage(_emp[pos].Imgstring);
("Male  if (_emp[pos].Gender == "
Male_rdbtn.Checked = true;
    else
       Female_rdbtn.Checked = true;
}
```

A Function for Converting an Image into a Textual Series to be Easily Saved in an XML File:

```
private static string ImageToString(Image img)
{
  var m = new MemoryStream();
 img.Save(m,  System.Drawing.Imaging.ImageFormat.Jpeg);
 return Convert.ToBase64String(m.ToArray());
}
```

A Function for retrieving an image from its textual format:

```
private static Image StringToImage(string imgstring)
{
var imgbytes = Convert.FromBase64String(imgstring);
 var m = new MemoryStream(imgbytes);
   return Image.FromStream(m);
}
```

A Code for Creating an XML File and Adding a Node to the File:

```
private void write_btn_Click(object sender, EventArgs e)
{
var found = false;
if (File.Exists(Xpath))
{
_xel = XElement.Load(Xpath);
foreach (var t in _emp)
{
found = t.ID.ToString(CultureInfo.InvariantCulture) == id_txt.Text;
}
}
else
_xel = new XElement("Info");
if (!found)
{
var gender = Male_rdbtn.Checked ? "Male" : "Female";
_empinfo = new XElement("EmpInfo");
_empinfo.SetAttributeValue("EmpInfo", "EmployeeInformation");
_id = new XElement("EId", id_txt.Text);
_name = new XElement("EName", name_txt.Text);
_name.SetAttributeValue("Gender", gender);
_age = new XElement("EAge", age_txt.Text);
_image = new XElement("EImage", _imgstr);
_empinfo.Add(_id, _name, _age, _image);
_xel.Add(_empinfo);
_xel.Save(Xpath);
_pos = _xel.Nodes().Count() - 1;
read_btn_Click(sender, e);
}
else
{
MessageBox.Show(@"This sequence is present ,choose another sequence ", "");
}
}
```

At this point, it is important to check if the file is available or not. If so, we read it only. Then, we make sure of the non-recurrence of the sequence for more than once and add the element to the file. If there is no file, then we create one file by creating its major root with the name Info and a

sub-node with the name EmpInfo to include the following nodes: ( ID, Name, Age, Image). While Name node has Gender as an attribute the Emp-Info node has an attribute as EmpInfo.

There are, for sure, various ways used in adding an element to an XML file. We have chosen XE-lement to be included in the field of XML.Linq names. This is the easiest way to create a file and add elements and the attributes of a specific ele-ment of those elements.

***Content Reading Code:***

```
private void read_btn_Click(object sender, Even-tArgs e)
{
srch_btn.Enabled = true;
_emp.Clear();
if (!File.Exists(Xpath)) return;
_doc.Load(Xpath);
var xmlElement = _doc[“Info”];
if (xmlElement != null)
_node = xmlElement[“EmpInfo”];
while (_node != null)
{
var element = _node[“EId”];
if (element != null)
{
var xmlElement1 = _node[“EName”];
if (xmlElement1 != null)
{
var element1 = _node[“EAge”];
if (element1 != null)
{
var xmlElement2 = _node[“EImage”];
if (xmlElement2 != null)
_emp.Add(new Emp
{
ID = int.Parse(element.InnerText),
Name = xmlElement1.InnerText,
Gender = xmlElement1.Attributes[“Gender”].In-nerText,
Age = int.Parse(element1.InnerText),
Imgstring= xmlElement2.InnerText ;
}
}
}
}
_node = _node.NextSibling;
}
Fill(_pos);      }
```

There are also various ways to read a file. Here, we use two objects, being XmlDocumnet and XmlNode, both of which are within the field of XML names. To do so, we first apply the main node and the node following it to the XmlNode variable object. That is, EmpInfo node is applied to the variable Node through which we pass to the node which it contains and extract its contents. Then, we move to each node by using NextSib-ling attribute, that is, the next node. Yet, reading a node content is done through InnerText attribute after mentioning the name of the node. To reach the attribute of a specific node, we first mention the name of the node followed by the name of the attribute, and then InnerText. As you may see, it is a very simple to reach the content of any node in a file.

**Content Update Code:**

```
private void update_btn_Click(object sender, EventArgs e)
{
var xmlElement = _doc[“Info”];
if (xmlElement != null)
_node = xmlElement[“EmpInfo”];
while (_node != null)
{
if (_node[“EId”] != null && _node[“EId”].Inner-Text == id_txt.Text)
{
_emp[_pos].ID = int.Parse(_node[“EId”].Inner-Text = id_txt.Text);
_emp[_pos].Name = _node[“EName”].InnerText = name_txt.Text;
if (Male_rdbtn.Checked)
{
_emp[_pos].Gender = _node[“EName”].Attri-butes[“Gender”].InnerText = Male_rdbtn.Text;
}
else
{
_emp[_pos].Gender = _node[“EName”].Attri-butes[“Gender”].InnerText = Female_rdbtn.Text;
}
_emp[_pos].Age = int.Parse(_node[“EAge”].In-nerText = age_txt.Text);
```

```
_emp[_pos].Imgstring = _node["EImage"].InnerText = ImageToString(pic.Image);
}
_node = _node.NextSibling;
}
_doc.Save(Xpath);
read_btn_Click(sender, e);
}
```

***Deletion Code:***

```
private void delete_btn_Click(object sender, EventArgs e)
{
var xmlElement = _doc["Info"];
if (xmlElement != null)
_node = xmlElement["EmpInfo"];
while (_node != null)
{
var element = _node["EName"];
if (element != null && element.InnerText == name_txt.Text)
{
if (_node.ParentNode != null) _node.ParentNode.RemoveChild(_node);
_emp.RemoveAt(int.Parse(id_txt.Text) - 1);
}
_node = _node.NextSibling;
}
_doc.Save(Xpath);
_pos--;
if (_pos < 0)
_pos = 0;
Fill(_pos);
}
```

We delete a specific node according to the name by using this line:

```
node.ParentNode.RemoveChild(node);
```

The above line deletes EmpInfo node which belongs to the main node Info. When Empinfo node is deleted, all other embedded nodes will be deleted including their contents or elements.

**The Search Code:**

```
private void srch_btn_Click(object sender, EventArgs e)
{
if (!File.Exists(Xpath)) return;
_doc.Load(Xpath);
var xmlElement = _doc["Info"];
if (xmlElement != null) _node = xmlElement["EmpInfo"];
while (_node != null)
{
if (_node["EName"].InnerText == Srch_txt.Text.Trim())
{
id_txt.Text = _node["EId"].InnerText;
name_txt.Text = _node["EName"].InnerText;
string gender = _node["EName"].Attributes["-Gender"].InnerText;
age_txt.Text = _node["EAge"].InnerText;
pic.Image = StringToImage(_node["EImage"].InnerText);
_pos = int.Parse(id_txt.Text) - 1;
if (gender == "ركذ")
Male_rdbtn.Checked = true;
else
Female_rdbtn.Checked = true;
}
_node = _node.NextSibling;
}}
```

***Reporting Code:***

```
private void button2_Click(object sender, EventArgs e)
{
var rf = new reportform();
rf.rv.LocalReport.EnableExternalImages = true;
var ds = new Microsoft.Reporting.WinForms.ReportDataSource("emp", _emp);
var param = new List<Microsoft.Reporting.WinForms.ReportParameter>
{
new Microsoft.Reporting.WinForms.ReportParameter("id",
_emp[_pos].ID.ToString(CultureInfo.InvariantCulture)),
new Microsoft.Reporting.WinForms.ReportParameter("name", _emp[_pos].Name),
new Microsoft.Reporting.WinForms.ReportParameter("age",
_emp[_pos].Age.ToString(CultureInfo.InvariantCulture))
};
var im = StringToImage(_emp[_pos].Imgstring);
const string temp = "c:\\Pics\\temp.jpg";
```

```
im.Save(temp);
param.Add(new Microsoft.Reporting.WinForms.
ReportParameter("image", "file:///c:\\Pics\\temp.
jpg"));
rf.rv.LocalReport.DataSources.Add(ds);
rf.rv.LocalReport.SetParameters(param);
rf.ShowDialog();
File.Delete(temp);        }
```

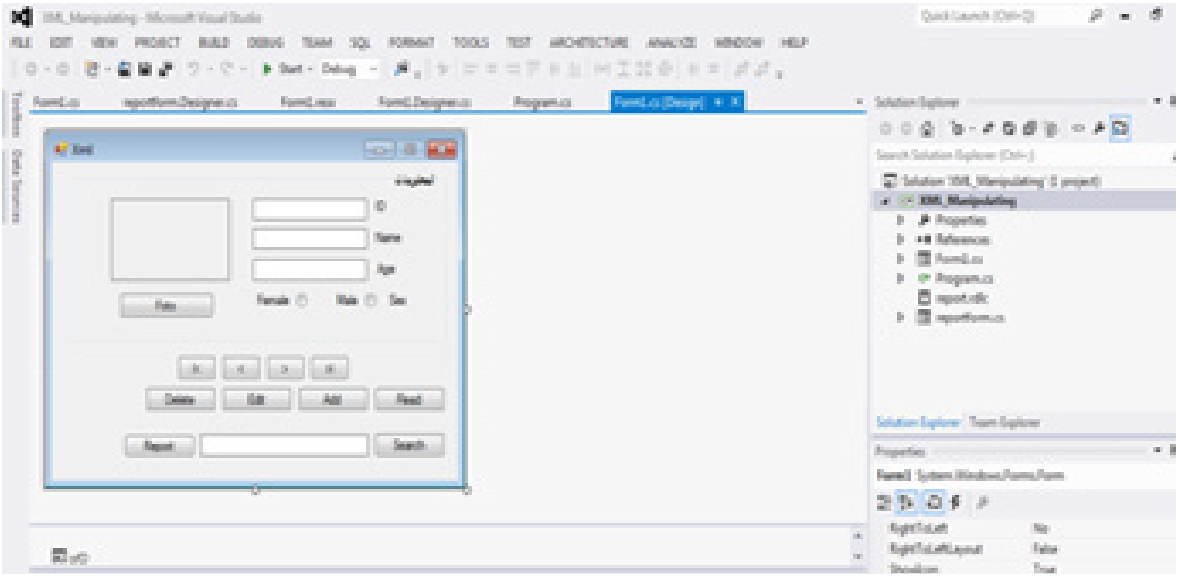


*Fig. 1: Illustrating a program in C# by using (NoSQL) XML database*

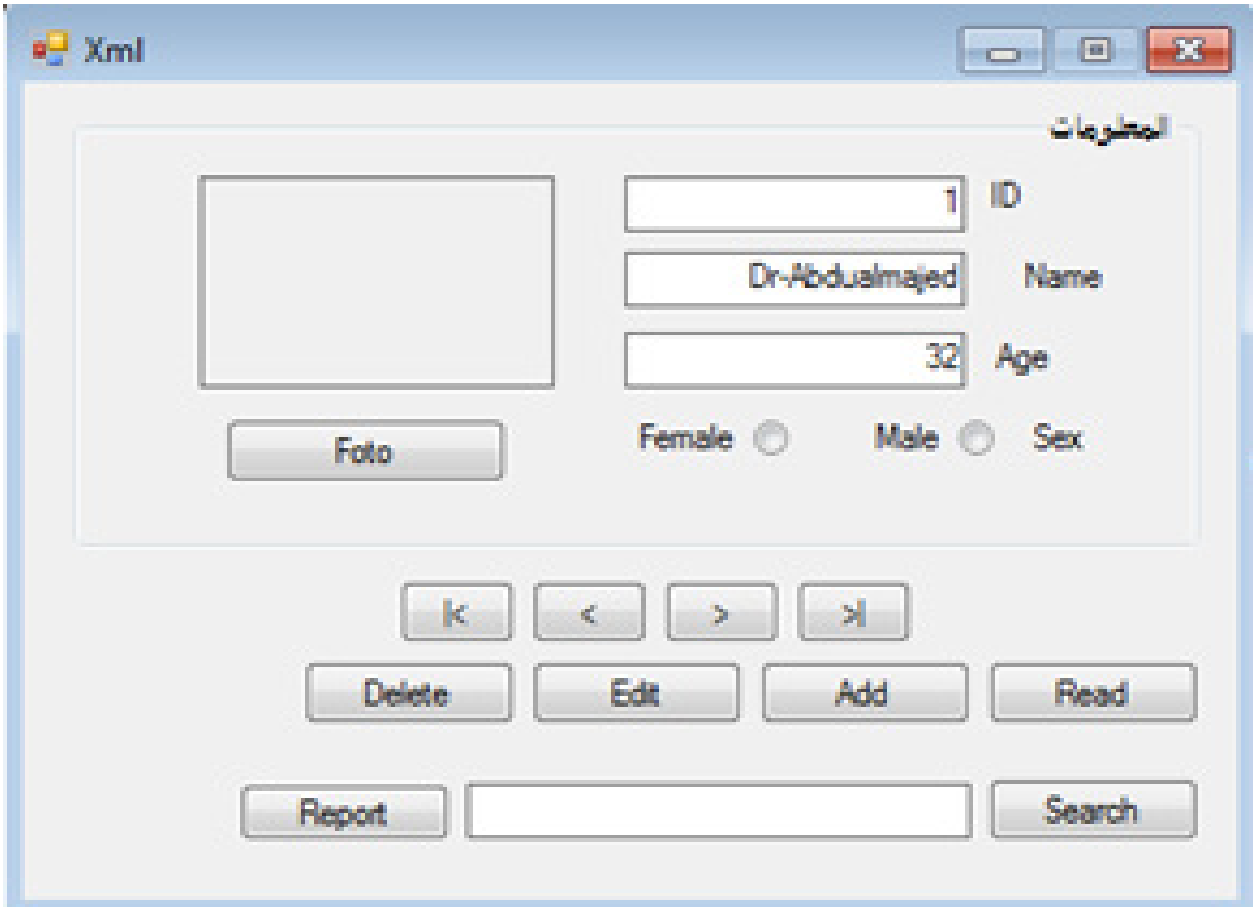



*Fig. 2: Implementation of the Program in C# by using (NoSQL) XML database*

Just like XML database, an NoSQL database could deal with Asp.net in terms of designing and developing dynamic websites instead of dealing with SQL database such as Oracle, Mysql, and Mssql server.

## 4.Comparing NoSQL Database with SQL Database

Before drawing a comparison between NoSQL databases and SQL databases, let's take a look at the different database management systems.


*Fig. 3: Showing the Way of Comparing NoSQL and SQL Databases*

### *4.1 Relational Database Management Systems (SQL)*

Relational Database Systems took its name after the model it is based on -The Relational Model, which was discussed earlier. These systems are and will remain for quite some time the best option to keep data reliable and safe. Not only that, they are also efficient. Relational database management systems require defined and clearly set schemas.

These schemas are much like tables; columns to contain a certain amount of information and to present the type of information in each record in addition to rows presenting the inputs.

Some of the most common relational database management systems include the followings [11]:

• SQLite: A very powerful and embedded relational database management system.

• MySQL: The most popular and commonly used RDBMS.

•PostgreSQL: The most advanced, SQL-compliant and open-source objective-RDBMS.

### *4.2 NoSQL Database Systems*

NoSQL database systems do not come with a model as the one used (or needed) by structured relational solutions. There are many applications; each application operates very differently and serves specific needs. Either these schema-less solutions allow an unlimited formation of inputs or entries, rather take, a very simple form but extremely efficient for operating as useful key based value stores[12,13].

NoSQL databases do not have a common way to query the data (i.e. similar to SQL of relational databases) and each solution provides its own query system.

Examples of NoSQL databases are Jackrabbit Mongodb, XML, Riak, CouchDB, and Cassandra.

| Term | Matching Databases |
|---|---|
| Data-Structures Server | Redis |
| Tuple Store | Gigaspaces<br>Coord<br>Apache River |
| Object Database | ZopeDB<br>DB4O<br>Shoal |
| Document Store | CouchDB<br>Mongo<br>Jackrabbit<br>XML Databases<br>ThruDB<br>CloudKit<br>Perservere<br>Riak Basho<br>Scalaris |
| Wide Columnar Store | Bigtable<br>Hbase<br>Cassandra<br>Hypertable<br>KAI<br>OpenNeptune<br>Qbase<br>KDI |

*Fig. 4: Classifications NoSQL Database*

- MongoDB: Cross-platform document-oriented database system that eschews the traditional table-based relational database structure in favor of JSON-like documents with dynamic schemas making the integration of data in certain types of applications faster and easier [10].

- Cassandra: Highly scalable, high performance distributed database designed to handle large amounts of data across many commodity Serv-

ers, and that provide high availability with no single point of failure [14].

- XML database is a data persistence software system that allows data to be specified, and sometimes stored, in XML format. These data can be queried, transformed, exported and returned to a calling system [3]. XML databases are a flavor of document-oriented databases which are in turn a category of NoSQL database (meaning Not (only) SQL).

### *4.3 A Comparison of SQL and No-SQL Database Management Systems*

In order to reach a simpler, understandable conclusion, let's analyze the differences between both SQL and No-SQL database management systems:

#### *4.3.1 Structure and Type of Data Being Kept:*

SQL/Relational databases require a structure with defined attributes to maintain or keep the data, unlike NoSQL databases which usually allow free-flow operations.

#### *4.3.2 Querying:*

Regardless of their licenses, all relational databases apply the SQL standard to a certain degree and thus, can be queried using the Structured Query Language (SQL). NoSQL databases, on the other hand, do not apply a unique way to operate the data they manage.

#### *4.3.3 Scaling:*

Both solutions are easy to scale vertically (i.e. by increasing a system's resources). However, being more modern (and simpler) applications, NoSQL solutions usually offer a much easier means to scale horizontally (i.e. by creating a cluster of multiple devices or machines).

#### *4.3.4 Reliability:*

When it comes to data reliability and safe guarantee of performed transactions, SQL databases are still the best option

#### *4.3.5 Support:*

Relational database management systems have a long history of strong performance and application. They are extremely common and their support is very easy to find both free of charge and/or paid. If an issue or problem arises, it is therefore much easier to solve such a problem than in the case of recently-popular NoSQL databases - especially if the said solution under focus is complex in nature (e.g. MongoDB).

#### *4.3.6 Complex data keeping and querying needs:*

By nature, relational databases are the ultimate solution for complex querying and data keeping needs. They are much more efficient and definitely excel in this domain.

### *4. 4The primary differences between SQL and NoSQL*

#### *4. 4.1SQL Tables and NoSQL Documents*

SQL databases provide a store of related data tables. For example, if you run an Online book store, book information can be added to a table named book:

| ISBN | Title | author | format | price |
|---|---|---|---|---|
| 9780992461225 | JavaScript: Novice to Ninja | Darren Jones | ebook | 29.00 |
| 9780994182654 | Jump Start Git | Shaumik-Daityari | ebook | 29.00 |

Every row is a different book record, the design is rigid; you cannot use the same table to store different information or insert a string where an integer is expected.

NoSQL databases store JSON-like field-value pair documents, e.g.

```
{
 ISBN: 9780992461225,
 title: "JavaScript: Novice to Ninja",
 author: "Darren Jones",
 format: "ebook",
 price: 29.00
}
```

Similar documents can be stored in a collection, which is analogous to an SQL table. However, you can store any data you like in any document;

the NoSQL database won't complain. For example:

```
{
 ISBN: 9780992461225,
 title: "JavaScript: Novice to Ninja",
 author: "Darren Jones",
 year: 2014,
 format: "ebook",
 price: 29.00,
 description: "Learn JavaScript from scratch!",
 rating: "5/5",
 review: [
  { name: "A Reader", text: "The best JavaScript book I've ever read." },
   { name: "JS Expert", text: "Recommended to novice and expert developers alike." }
}
```

SQL tables create a strict data template, so it's difficult to make mistakes. NoSQL is more flexible and forgiving, but being able to store any data anywhere which can lead to consistency issues.

### *4. 4.2SQL Schema and NoSQL Schema-less*

In an SQL database, it's impossible to add data until you define tables and field types in what's referred to as a schema. The schema optionally contains other information, such as —

- **Primary keys** — unique identifiers such as the ISBN which apply to a single record
- **Indexes** — commonly queried fields indexed to aid quick searching
- **Relationships** — logical links between data fields
- **Functionality** such as triggers and stored procedures.

Your data schema must be designed and implemented before any business logic can be developed to manipulate data. It is possible to make updates later, but large changes can be complicated.

In a NoSQL database, data can be added anywhere, at any time, there's no need to specify a document design or even a collection up-front, for example, in MongoDB the following statement will create a new document in a new book collection if it's not been previously created:

```
db.book.insert(
 ISBN: 9780994182654,
 title: "Jump Start Git",
 author: "ShaumikDaityari",
 format: "ebook",
 price: 29.00
);
```

(MongoDB will automatically add a unique _id value to each document in a collection. You may still want to define indexes, but that can be done later if necessary.)

A NoSQL database may be more suited to projects where the initial data requirements are difficult to ascertain. That said, don't mistake difficulty for laziness: neglecting to design a good data store at project commencement will lead to problems later.

### *4.4.3 SQL Normalization and NoSQL Denormalization*

Presume we want to add publisher information to our book store database, a single publisher could offer more than one title so, in an SQL database, we create a new publisher table:

| id | name | country | Email |
|---|---|---|---|
| SP001 | SitePoint | Australia | feedback@sitepoint.com |

We can then add a publisher_id field to our book table, which references records by publisher.id:

| ISBN | Title | author | format | price | publisher_id |
|---|---|---|---|---|---|
| 9780992461225 | JavaScript: Novice to Ninja | Darren Jones | ebook | 29.00 | SP001 |
| 9780994182654 | Jump Start Git | Shaumik Daityari | ebook | 29.00 | SP001 |

This minimizes data redundancy; we're not repeating the publisher information for every book only the reference to it. This technique is known

as normalization, and has practical benefits. We can update a single publisher without changing book data.

We can use normalization techniques in NoSQL. Documents in the book collection —

```
{
 ISBN: 9780992461225,
 title: “JavaScript: Novice to Ninja”,
 author: “Darren Jones”,
 format: “ebook”,
 price: 29.00,
 publisher_id: “SP001”
}
```

— reference a document in a publisher collection:

```
{
 id: “SP001”
 name: “SitePoint”,
 country: “Australia”,
 email: “feedback@sitepoint.com”
}
```

However, this is not always practical, for reasons that will become evident below. We may opt to denormalize our document and repeat publisher information for every book:

```
{ISBN: 9780992461225,
 title: “JavaScript: Novice to Ninja”,
 author: “Darren Jones”,
 format: “ebook”,
 price: 29.00,
 publisher: {
   name: “SitePoint”,
   country: “Australia”,
email: “feedback@sitepoint.com”}}
```

This leads to faster queries, but updating the publisher information in multiple records will be significantly slower.

### *4. 4.4 SQL Relational JOIN and NoSQL*

SQL queries offer a powerful JOIN clause. We can obtain related data in multiple tables using a single SQL statement. For example:

```
SELECT book.title, book.author, publisher.name
FROM book
LEFT JOIN book.publisher_id ON publisher.id;
```

This returns all book titles, authors and associated publisher names (presuming one has been set). NoSQL has no equivalent of JOIN, and this can shock those with SQL experience. If we used normalized collections as described above, we would need to fetch all book documents, retrieve all associated publisher documents, and manually link the two in our program logic, this is one reason denormalization is often essential.

### *4. 4.5 SQL and NoSQL Data Integrity*

Most SQL databases allow you to enforce data integrity rules using foreign key constraints (unless you’re still using the older, defunct MyISAM storage engine in MySQL). Our book store could
•ensure all books have a valid publisher_id code that matches one entry in the publisher table, and
•not permit publishers to be removed if one or more books are assigned to them.

the schema enforces these rules for the database to follow, it’s impossible for developers or users to add, edit or remove records, which could result in invalid data or orphan records.

The same data integrity options are not available in NoSQL databases; you can store what you want regardless of any other documents. Ideally, a single document will be the sole source of all information about an item.

### *4. 4.6 SQL and NoSQL Transactions*

In SQL databases, two or more updates can be executed in a transaction — an all-or-nothing wrapper that guarantees success or failure. For example, presume our book store contained order and stock tables, when a book is ordered we add a record to the orderTable and decrement the stock count in the stock table. If we execute those two updates individually, one could succeed and the other fail — thus leaving our figures out of sync, placing the same updates within a transaction ensures either both succeed or both fail.

In a NoSQL database, modification of a single document is atomic, in other words, if you’re updating three values within a document, either all three are updated successfully or it remains unchanged. However, there’s no transaction equivalent for updates to multiple documents, there are transaction-like options, but -at the time of writing this article- these must be manually processed in your code.

### *4. 4.7 SQL vs NoSQL CRUD Syntax*

Creating, reading updating and deleting data is the basis of all database systems. In essence —
•SQL is a lightweight declarative language. It’s

deceptively powerful, and has become an international standard, although most systems implement subtly different syntaxes.

•NoSQL databases use JavaScripty-looking queries with JSON-like arguments! Basic operations are simple, but nested JSON can become increasingly convoluted for more complex queries.

**Table 3 :** Comparison between SQL and NoSQL

| SQL | NoSQL |
|---|---|
| insert a new book record | |
| INSERT INTO book ( `ISBN`, `title`, `author`) VALUES ( '9780992461256', 'Full Stack JavaScript','Colin Ihrig& Adam Bretz'); | db.book.insert({ ISBN: "9780992461256", title: "Full Stack JavaScript", author: "Colin Ihrig& Adam Bretz"}); |
| update a book record | |
| UPDATE book SET price = 19.99 WHERE ISBN = '9780992461256' | db.book.update({ ISBN: 9780992461256' },{ $set: { price: 19.99 }); |
| return all book titles over $10 | |
| SELECT title FROM book WHERE price > 10; | db.book.find({ price: { >: 10 } }, { _id: 0, title: 1 }); The second JSON object is known as a projection: it sets which fields are returned (_id is returned by default so it needs to be unset). |
| count the number of SitePoint books | |
| SELECT COUNT(1) FROM book WHERE publisher_id = 'SP001'; | db.book.count({"publisher.name": "SitePoint"}); This presumes denormalized documents are used. |
| return the number of book format types | |
| SELECT format, COUNT(1) AS `total` FROM book GROUP BY format; | db.book.aggregate([{ $group: { _id: "$format", total: { $sum: 1 } }});This is known as aggregation: a new set of documents is computed from an original set. |
| delete all SitePoint books | |
| DELETE FROM book WHERE publisher_id = 'SP001'; Alternatively, it's possible to delete the publisher record and have this cascade to associated book records if foreign keys are specified appropriately. | db.book.remove({"publisher.name": "SitePoint"}); |

The second JSON object is known as a projection: it sets which fields are returned (_id is returned by default so it needs to be unset).

count the number of SitePoint books

SELECT COUNT(1) FROM book WHERE publisher_id = 'SP001'; db.book.count({ "publisher.name": "SitePoint"});

This presumes denormalized documents are used.

return the number of book format types

SELECT format, COUNT(1) AS `total` FROM book GROUP BY format; db.book.aggregate([ { $group: { _id: "$format", total: { $sum: 1 } }});This is known as aggregation: a new set of documents is computed from an original set.

delete all SitePoint books

DELETE FROM book WHERE publisher_id = 'SP001';

Alternatively, it's possible to delete the publisher record and have this cascade to associated book records if foreign keys are specified appropriately. db.book.remove({"publisher.name": "SitePoint"});

### *4. 5 SQL and NoSQL: High-Level Differences*

•SQL databases are primarily called as Relational Databases (RDBMS); whereas NoSQL database are primarily called as non-relational or distributed database.

•SQL databases are table based databases where as NoSQL databases are document based, key-value pairs, graph databases or wide-column stores, this means that SQL databases represent data in form of tables which consists of n number of rows of data whereas NoSQL databases are the collection of key-value pair, documents, graph databases or wide-column stores which do not have standard schema definitions which it needs to adhered to.

•SQL databases have predefined schema whereas NoSQL databases have dynamic schema for unstructured data.

•SQL databases are vertically scalable whereas the NoSQL databases are horizontally scalable; SQL databases are scaled by increasing the horse-power of the hardware. NoSQL databases are scaled by increasing the databases servers in the pool of resources to reduce the load.

•SQL databases use SQL ( structured query language ) for defining and manipulating the data which is very powerful; In NoSQL databases queries are focused on collection of documents, sometimes it is also called UnQL (Unstructured Query Language). The syntax of using UnQL varies from database to database.

•SQL database examples: MySql, Oracle, SQLite, Postgres and MS-SQL. NoSQL database examples: MongoDB, BigTable, Redis, RavenDb, Cassandra, Hbase, Neo4j and CouchDb.

•**For complex queries:** SQL databases are a good fit for the complex query intensive environment whereas NoSQL databases are not good fit for complex queries. On a high-level, NoSQL don't have standard interfaces to perform complex queries, and the queries themselves in NoSQL are not as powerful as SQL query language.

•**For the type of data to be stored:** SQL databases are not best fit for hierarchical data storage, where NoSQL database fits better for the hierarchical data storage as it follows the key-value pair way of storing data similar to JSON data. NoSQL database are highly preferred for large data set (i.e for big data). Hbase is an example for this purpose.

•**For scalability:** In most typical situations, SQL databases are vertically scalable. You can manage increasing load by increasing the CPU, RAM, SSD, etc, on a single server. On the other hand, NoSQL databases are horizontally scalable. You can just add few more servers easily in your NoSQL database infrastructure to handle the large traffic.

•**For high transactional based application:** SQL databases are best fit for heavy duty transactional type applications, as it is more stable and promises the atomicity as well as integrity of the data. While you can use NoSQL for transactions purpose, it is still not comparable and sable enough in high load and for complex transactional applications.

•**For support:** Excellent support are available for all SQL database from their vendors, there are also lot of independent consultations who can help you with SQL database for a very large scale deployments. For some NoSQL database you still have to rely on community support, and only limited outside experts are available for you to setup and deploy your large scale NoSQL deployments.

•**For properties:** SQL databases emphasizes on ACID properties ( Atomicity, Consistency, Isolation and Durability) whereas the NoSQL database follows the Brewers CAP theorem ( Consistency, Availability and Partition tolerance )

•**For DB types:** On a high-level, we can classify SQL databases as either open-source or close-sourced from commercial vendors. NoSQL databases can be classified on the basis of way of storing data as graph databases, key-value store databases, document store databases, column store database and XML databases

## 5. Conclusion

It is safe to argue that NoSQL represents a very powerful and new means for data storage and retrieval in an optimized and smooth manner. Applying NoSQL is easier to deal with data than using SQL in this regard.

Criticizing any one of the SQL's will not help the aim of this work. If there is a buzz of NoSQL these days, it doesn't mean that is a silver bullet to all your needs. Both technologies (SQL and NoSQL) are best in what they do. It is up to a

developer to make a better use of them depending on the situations and needs.

NoSQL databases, by using an unstructured (or structured) kind of approach, aim to eliminate the limitations of strict relations, and accordingly, offer many different ways to maintain and efficiently use data for specific usage cases (e.g. full-text document storage).

## References

[1] XML Media Types, RFC 7303. Internet Engineering Task Force. July 2014.
[2] XML 1.0 Specification. World Wide Web Consortium. Retrieved 2010-08-22.
[3] XML and Semantic Web W3C Standards Timeline (PDF). 2012-02-04.
[4] Fennell, Philip (June 2013). "Extremes of XML". XML London 2013: 80–86.doi:10.14337/XMLLondon13.Fennell01. ISBN 978-0-9926471-0-0.
[5]M. Murata, D. Kohn, and C. Lilley (2009-09-24). "Internet Drafts: XML Media Types". Internet Engineering Task Force. Retrieved 2012-02-29.
[6] Extensible Markup Language (XML) 1.1 (Second Edition) . World Wide Web Consortium. Retrieved 2010-08-22.
[7] Pilgrim, "The history of draconian error handling in XML". Archived from the original on 2011-07-26. Retrieved 18 July 2013.
[8] Jon Bosak, "Closing Keynote, XML 2006". 2006.xmlconference.org. Archived from the original on 2007-07-11. Retrieved2009-07-31.
[9] T. Bray, C. Frankston, A. Malhotra, "Document Content Description for XML", http://www.w3.org/TR/NOTE-dcd.
[10] Obasanjo, D. (2013). Building scalable databases: Denormalization, the NoSQL movement and Digg. Retrieved July 15th.
[11] Software Engineering Radio Podcast, Episode 165: NoSQL and MongoDB with Dwight Merriman by Robert Blumen and Dwight Merriman: http://www.se-radio.net/2010/07/episode-165-nosql-and-mongodb-with-dwight-merriman/
[12] Heise SoftwareArchitekTOUR Podcast (German), Episode 22: NoSQL – Alternative zu relationalen Datenbanken by Markus Völter, Stefan Tilkov and Mathias Meyer: http://www.heise.de/developer/artikel/Episode-22-NoSQL-Alternative-zu-relationalen-Datenbanken- 1027769.htm.
[13] Zawodny, J. (2009). NoSQL: Distributed and Scalable Non-Relational Database Systems. Linux Magazine web portal, http://www. linux-mag. com/id/7579.
[14] RadioTux Binärgewitter Podcast (German), Episode 1: NoSQL by Dirk Deimeke, Marc Seeger, Sven Pfleiderer and Ingo Ebel: http://blog.radiotux.de/2011/01/09/binaergewitter-1-nosql

journal homepage: www.sabauni.net/ojs

Saba Journal of Information Technology and Networking (SJITN)

# Article

## Measure Security Requirement of Critical System Using Fuzzy Logic

**Dr. Hazaa M. *, Gassem,Y. , Alzidani, M. , Alansi, S. , Albadani, A.**

*Thamar University, Yemen*

**Article info**

**Article history:**

Accepted Dec, 2016

**Keywords:**

Security requirement specification ,
Fuzzy logic ,
Critical system ,
Threat

**Abstract**

Security is one of the important dependable system dimensions, as it plays a primary role in a critical system that must be dependable and provides acceptable degrees of security, safety and integrity. To achieve success in this type of system development, and to avoid loss and failure of the system that we want to develop, it is important to be sure that the system specification that relates to the security is correct and can be developed in organization's abilities and that the required security level for the requirements specification of the critical system is acceptable to the organization, and can be developed in terms of costs, technology and assets. For this purpose, we will build a method that will perform the analysis, support and confidence ensuring that these requirements are acceptable and whether it can be applied in the system or not. Our method will use fuzzy logic to capture knowledge from analysis experts as rules that would help take a certain decision with respect to asset values, available technology and threats of the organization. From this, our methodology will give the analyst a level of confidence and acceptance for the security Requirement specification that we deal with.

* Corresponding author: Dr.Muneer Hazaa
E-mail: muneer_hazaa@yahoo.com

## 1. Introduction

Any critical system must always be secure. The systems that are not secure will not be dependable, and consequently will affect other dependability factors like availability, reliability and safety [1]. So, it is important to build a plan and test the software process specification's requirements before we build such a type of critical system that can be applied, and to be sure to some degree of certainty that the requirements are acceptable and applicable [16] .

In fact, the security depends on many factors which in most cases cannot be covered because of the spread of security domain. Also, types of attacks and technology that can be implemented in this kind of system are variable. Furthermore, there are several processes for identifying and prioritizing risks. One of the most effective processes is threat modeling. Threat modeling is the process of identifying, quantifying and analyzing potential threats of a computer-based system. It is a process of assessing and documenting a system's security risks [4]. It involves identifying the key assets of an application, decomposing the application, identifying and categorizing the threats to each assets or component, rating the threats based on a risk ranking, and then developing threat mitigation strategies that are then implemented in design, code and test cases [5]. Categorizing threats is the first step toward effective mitigation [4].Threats can be classified into six classes based on their effect [6]. This is generally referred to as the STRIDE model. The STRIDE (Spoofing, Tampering, Repudiation, Information disclosure, Denial of Service and Elevation of privilege) model was used by Microsoft for categorizing threats [5]. Security measurements can be achieved by estimating the ability of building these requirements that we deal with in the specification phase. This method will be useful to support the Analyst who doesn't have enough experience with such kind of system and help support to achieve the optimal level of certainty regarding these requirements.

### 2. Security Requirements with fuzzy logic

When security requirements are considered at requirements specification stage writing from the system development life cycle, they tend to be general lists of security features, such as password protection, firewalls, virus detection tools, and the like [10]. These are, in fact, not security requirements at all but rather implementation mechanisms that are intended to satisfy unstated requirements, such as authenticated access. As a result, security requirements that are specific to the system and that provide protection of essential services and assets are often neglected. In reviewing requirements documents, we typically find that security requirements, when they exist, are in a section by themselves and have been copied from a generic set of security requirements. The requirements elicitation and analysis, needed to get a better set of security requirements, seldom take place and measure these requirements if it will achieve the level of security required [11]. A proposed model consists of three steps that generate a numerical value in which the degree of security available appears and helps to detect prioritized security requirements and accept or reject this system depending on measurement value.

## 3. Literature Review

Security is a system attribute that reflects the ability of the system to protect itself from external attacks that may be accidental or deliberate [1]. The specialized terminologies associated with security are exposure, vulnerability, attack, threats and control [2].The assessment of system security is becoming increasingly important as more and more critical systems are Internet-enabled and can be accessed by anyone with a network connection [1]. These types of security assessment are very difficult to carry out. Consequently, systems are often deployed with security loopholes that attackers use to gain access to or damage these systems [3]. It is very difficult for end-users of a system to verify its security. Consequently, bodies in North America and Europe have established sets of security evaluation criteria that can be checked by specialized evaluators. Software product suppliers can submit their products for evaluation and certification against these criteria [7]. Therefore, if you have a requirement for a particular level of security, you can choose

a product that has been validated to that level [1]. However, many products are not security-certified or their certificate applies only to individual products. When the certified system is used in conjunction with other uncertified systems, such as locally developed software, the security level of the overall system cannot be assessed [1].

Fuzzy Logic introduced by Zadeh (1965) gives us a language, with syntax and local semantics, in which we can translate our qualitative knowledge about the problem to be solved [8]. Fuzzy logic is a powerful problem-solving methodology with a myriad of applications in embedded control and information processing [14]. Fuzzy logic provides a remarkably simple way to draw definite conclusions from vague, ambiguous or imprecise information. In a sense, fuzzy logic resembles human decision making with its ability to work from approximate data and find precise solutions. There are many factors which account for the increase in question but the most prominent among them is the rapidly growing use of soft computing and especially fuzzy logic in the conception and design of intelligent systems. As one of the principal constituents of soft computing, fuzzy logic is playing a key role in the conception and design of various systems [15]. There are two concepts within fuzzy logic which play a central role in its applications. The first is that of a linguistic variable, i.e., a variable whose values are words or sentences in a natural or synthetic language. The other is that of a fuzzy if-then rule in which the antecedent and consequent are propositions containing linguistic variables [8]. The essential function served by linguistic variables is that of granulation of variables and their dependencies. In effect, the use of linguistic variables and fuzzy if-then rules results -through granulation in soft data compression which exploits the tolerance for imprecision and uncertainty.

In fact, the theory of fuzzy sets theory is a generalization and extension of conventional nature which agrees with the language and understanding of human nature as well [13].

**Definition 1**.Suppose that X is a set of reference, the common characteristic of a subset A of X, is defined as follows:

$$\mu_A(x) = \begin{cases} 1 & : \ x \in A \\ 0 & : \ x \notin A \end{cases} \qquad (1)$$

According to the above definition, for each X Ɛ x, it will be only one of the values 0 or 1.

**Definition 2.** If the range of the function μA of the [1, 0] to the interval [1, 0] expands, we have a function to every member of X, the number in the range [1, 0] assigns. The other set A is not normal, but is called a fuzzy set (A is a fuzzy subset of X). In the above definition, if μA (X) Ɛ [0,1] then the membership of x belongs to variable A, with a certain degree between [0,1]. In fact, here is an extended concept of membership of an element. It also represents the membership degree of μA (X), the membership in a fuzzy set is the element x. If the degree of membership of an element is set to zero, the member is fully withdrawn [9]. And if the degree of membership of a member is set to one, the member is quite a collection. If the degree of membership of a member is between zero and one, the number that indicates the degree of membership is gradual. Figure 1 is an example of the membership function of a fuzzy set.

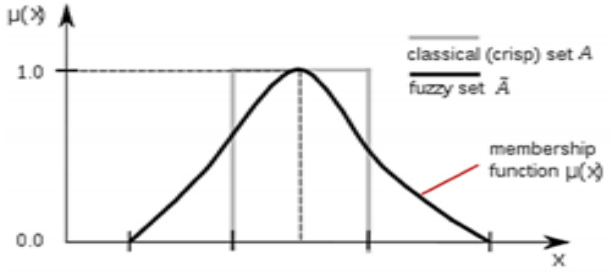



*Fig. 1.Membership Function of a Fuzzy Set*

In this paper we use a Gaussian membership function. A Gaussian membership function can be demonstrated by the following equation:

$$\mu_{A^i}(x) = \exp(-\frac{(c_i - x)^2}{2\sigma_i^2})$$

### *3.1 Fuzzy logic operation steps are described as follows:*

**Step1.Fuzzy Inputs:**

The first step is to take the inputs and determine the degree to which they belong to each of the appropriate fuzzy sets via membership functions. In Fuzzy Logic Toolbox™ software, the input is

always a crisp numerical value limited to the universe of discourse of the input variable (in this case the interval between 0 and 10) and the output is a fuzzy degree of membership in the qualifying linguistic set (always the interval between 0 and 1). Fuzzification of the input amounts to either a table lookup or a function evaluation [12].

**Step2.Apply Fuzzy Operator:**

After the inputs are fuzzified, you know the degree to which each part of the antecedent is satisfied for each rule. If the antecedent of a given rule has more than one part, the fuzzy operator is applied to obtain one number that represents the result of the antecedent for that rule. This number is then applied to the output function. The input to the fuzzy operator is two or more membership values from fuzzified input variables. The output is a single truth value [12].

**Step3.Apply Implication Method:**

Before applying the implication method, you must determine the rule's weight. Every rule has a weight (a number between 0 and 1), which is applied to the number given by the antecedent. Generally, this weight is 1 (as it is for this example) and thus has no effect at all on the implication process. From time to time you may want to weight one rule relative to the others by changing its weight value to something other than 1. After proper weighting has been assigned to each rule, the implication method is implemented. A consequent is a fuzzy set represented by a membership function, which appropriately weights the linguistic characteristics that are attributed to it. This consequent is reshaped using a function associated with the antecedent (a single number). The input for the implication process is a single number given by the antecedent, and the output is a fuzzy set. Implication is implemented for each rule. Two built-in methods are supported, and they are the same functions that are used by the AND method: min (minimum), which truncates the output fuzzy set, and prod (product), which scales the output fuzzy set. [12].

**Step4.Aggregate All Outputs:**

Because decisions are based on the testing of each rule in an FIS, the rules must be combined in some manner in order to make a decision. Aggregation is the process by which the fuzzy sets that represent the outputs of each rule are combined into a single fuzzy set. Aggregation only occurs once for each output variable, just prior to the fifth and final step, defuzzification. The input of the aggregation process is the list of truncated output functions returned by the implication process for each rule. The output of the aggregation process is one fuzzy set for each output variable [12].

**Step5.Defuzzify:**

The input for the defuzzification process is a fuzzy set (the aggregate output fuzzy set) and the output is a single number. As much as fuzziness helps the rule evaluation during the intermediate steps, the final desired output for each variable is generally a single number. However, the aggregate of a fuzzy set encompasses a range of output values, and so must be defuzzified in order to resolve a single output value from the set [12].

## 4. Design Methodology

In this section, we will explain the proposed methodology to determine the possibility of applying security requirements according to the specification criteria which is related to the security requirements in the critical systems, where the criteria which we use in the methodology is Assets in the system and the possible or expectant threats on this assets. The whole task starts with identifying the scope of the product. Asset based risk management is then conducted to identify the risks for all critical assets. Critical assets are identified based on costs of production and reproduction, the amount of loss for any damages, etc. Possible threats and vulnerabilities to this critical asset are then determined through threat profiles, attack trees, threat sources etc. Identification of asset, threat and vulnerability related to these assets are critical elements for risk identification. These risks are then analyzed by the likelihood of occurrence and by estimating their negative impact. The negative impact can be computed according to proportions between 1 and 100. If this proportion is low, that means that it has a low risk impact, and if the result is of a high value, then the result will be a high risk impact. Finally,

a mitigation plan, protection strategies and action lists are suggested to control the risk at an acceptable level. Security goals and policies are then outlined considering the product and organization. Security goals are the organization's motivation and business gain by applicability of the management control principles. Security policy sets out conditions to achieve the security goals

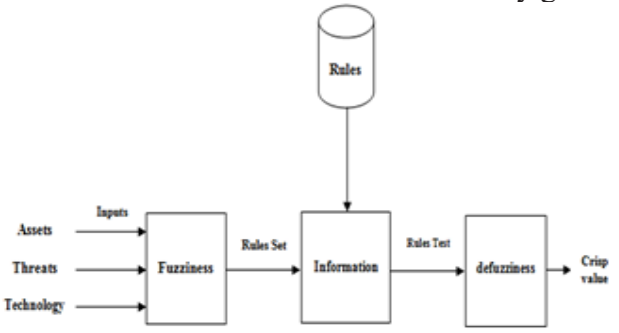



*Figure 2: Proposed Method Architecture for Fuzzy Logic-based Assets, Threat and Technology Modeling*

**The steps involved in the design are:**

**First. Fuzzification Phase:**

- Determining the cost of the assets (Memory, Operating System, Internet Explorer, Wired Cable, Sensor, Staff and Web Site) that are related to software and system. Here, we mean by assets, data and application. The value of the assets is determined according the system analysts and their experience or knowledge on the assets. Assets will determine the level of costing which may be cheap, normal, expensive or very expensive.

**2. Linguistic variable: Assets**

*Table 1. The input linguistic variable (Assets)*

| Linguistic value | Numerical range |
|---|---|
| Cheap | [-0.3333 0 0.3333] |
| Normal | [0 0.3333 0.6667] |
| Expensive | [0.3333 0.6667 1] |
| Very-Expensive | [0.6667 1 1.333] |

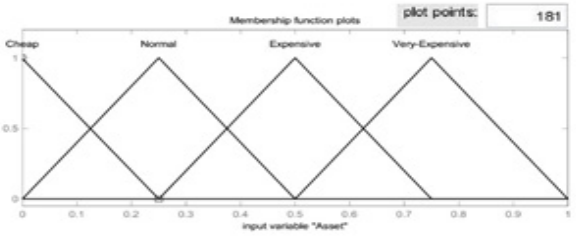



*Fig. 3: Assets Value on Degree of Membership*

- Determining the possible threats (Attacker , Resist, Damage , Risk , Unwanted properties , Constraint , Follow register , Exponent , Exposure , Denial of service , Mistakes ,Omission ,Vulnerability , maintenance ,install , download, configuration ,updating , number version , reparability, response , design , number activities within program, monitor attack , determining , survivability , reparability, environments) on this asset and determining the level of hazardousness. Also, the system analyst determines the level of hazardousness of threats for this requirement which may be: Intolerable , as low as practical , acceptable.

*Table 2. The input linguistic variable (Threats)*

| Linguistic value | Numerical range |
|---|---|
| Low | [-0.4 0 0.4] |
| Medium | [0.1 0.5 0.9] |
| High | [0.6 1 1.4] |

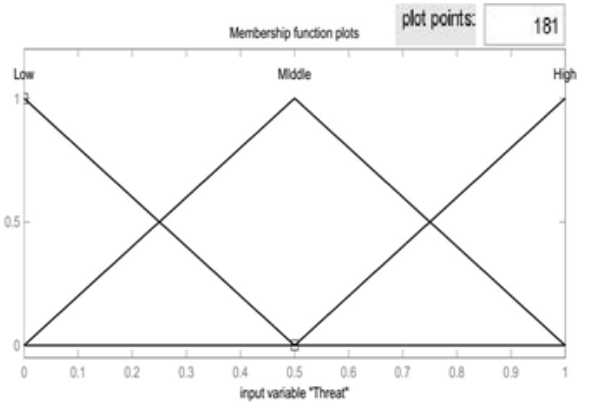



*Fig. 4: Threats Value on Degree of Membership*

-Determining the existing techniques (Antivirus, protocols, validation tools, verification tools, test tools, authentication, session use, encryption tools, documentation, password, applications and Operating System) in the markets and their costs.

*Table 3. The input linguistic variable (Technology)*

| Linguistic value | Numerical range |
|---|---|
| Easy-to-get | [-50 0.3 0.7] |
| Hard-to-get | [0.25 0.7 50] |

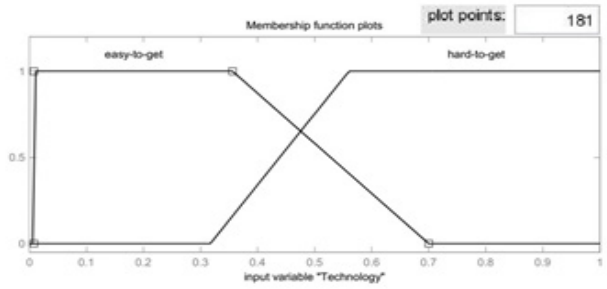



*Figure 5: Technology Value on Degree of Membership*

- These steps (1,2,3) are the inputting phase for the methodology under the name Fuzzification, where they will transform form the verbal variable for the inputting of assets, threats and technology into variables for the methodology or following

**Second: Rule and Inference Phase:**

In the Phase of Rule and inference, the output of the Fuzzification phase is applied to the input linguistic variable assets. Technology and threats apply the rules on these variables. After that the output of each linguistic variables is applied on mamdani inference function to get the result of the risk.. Based on these results we can determine the level of possibility of applying this requirement or not.

*Table 4. Samples of rules*

| Rules |
|---|
| 1. If (Technology is Easy-to-get) and (Threats is Low) and (Assets is Very-Expensive) then (Level-of-Requirement-Risk is Acceptable) |
| 2. If (Technology is Easy-to-get) and (Threats is Low) and (Assets is Normal) then (Level-of-Requirement-Risk is Acceptable) |
| 3. If (Technology is Easy-to-get) and (Threats is Low) and (Assets is Expensive) then (Level-of-Requirement-Risk is ALARP) (1) |
| 4. If (Technology is Easy-to-get) and (Threats is Low) and (Assets is Very-Expensive) then (Level-of-Requirement-Risk is ALARP) |
| 5. If (Technology is Easy-to-get) and (Threats is Medium) and (Assets is Cheap) then (Level-of-Requirement-Risk is Acceptable) (1) |
| 6. If (Technology is Easy-to-get) and (Threats is Medium) and (Assets is Normal) then (Level-of-Requirement-Risk is ALARP) (1) |
| 7. If (Technology is Easy-to-get) and (Threats is Medium) and (Assets is Expensive) then (Level-of-Requirement-Risk is ALARP) (1) |
| 8. If (Technology is Easy-to-get) and (Threats is High) and (Assets is Cheap) then (Level-of-Requirement-Risk is ALARP) (1) |
| 9. If (Technology is Easy-to-get) and (Threats is High) and (Assets is Normal) then (Level-of-Requirement-Risk is ALARP) (1) |
| 10. If (Technology is Easy-to-get) and (Threats is High) and (Assets is Expensive) then (Level-of-Requirement-Risk is Unacceptable) (1) |
| 11. If (Technology is Hard-to-get) and (Threats is Low) and (Assets is Cheap) then (Level-of-Requirement-Risk is ALARP) (1) |
| 12. If (Technology is Hard-to-get) and (Threats is Low) and (Assets is Normal) then (Level-of-Requirement-Risk is ALARP) (1) |
| 13. If (Technology is Hard-to-get) and (Threats is Low) and (Assets is Expensive) then (Level-of-Requirement-Risk is Unacceptable) (1) |
| 14. If (Technology is Hard-to-get) and (Threats is Medium) and (Assets is Expensive) then (Level-of-Requirement-Risk is Unacceptable) (1) |
| 15. If (Technology is Hard-to-get) and (Threats is High) and (Assets is Expensive) then (Level-of-Requirement-Risk is Unacceptable) (1) |

**Third: Defuzzification Phase:**

After the inference rule phase is finished, which has been handled based on the input related to the security requirement in Fuzzification phase, the Defuzzification will transform and output the result of inference to a specific range that represents the risk of developing this requirement and find out whether the requirement is accepted or not according to the criteria mentioned above.

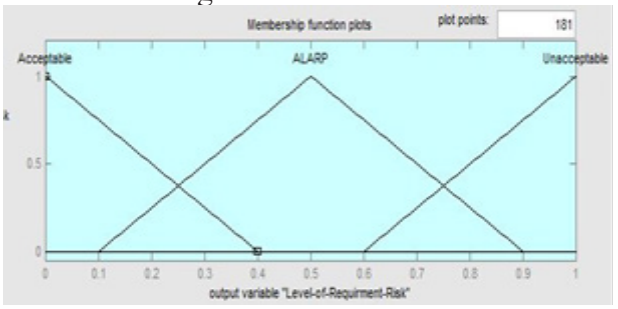



*Fig.6: Risk Value on Degree of Membership*

*Table 5. The output linguistic variable*

| Linguistic value | Numerical range |
|---|---|
| Cheap | [-0.3333 0 0.3333] |
| Normal | [0 0.3333 0.6667] |
| Expensive | [0.3333 0.6667 1] |
| Very-Expensive | [0.6667 1 1.333] |

## 5. Implementation and Evaluation

The methodology is implemented using MATLAB fuzzy logic toolbox. Implementations are presented below:

### 1- FIS Editor (Figure 6):

This window is used to select a new FIS type with any particular model, we can add the related variables to the mode, and input or output variable names can be added as well. In our methodology, we chose the model of Mamdani.

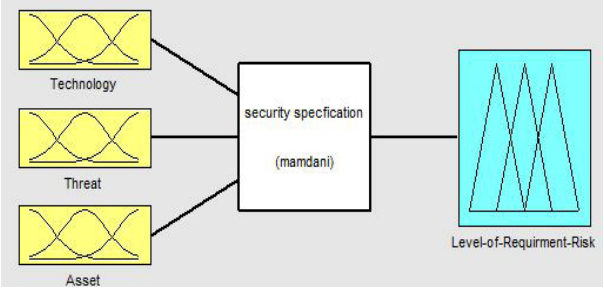



*Fig. 7: FIS Editor*

### 2- Membership function editor (Fig. 3 and 4):

This window is used for the input or the output of the membership function that can be added or removed. It also makes it possible to specify the ranges of each of the variables and membership functions.

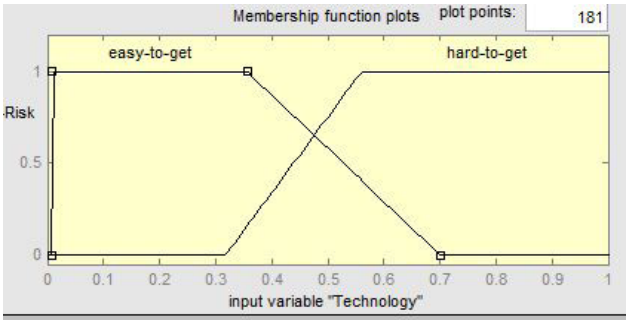



*Fig. 8: Technology Input Variable*

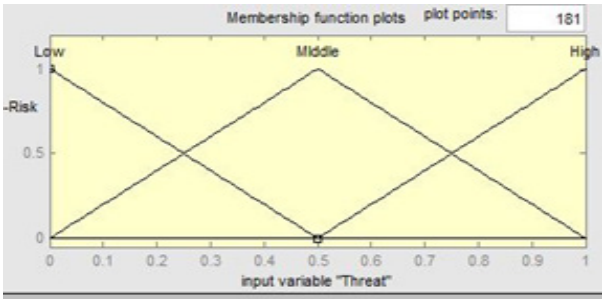



*Fig. 9: Threat Input Variable*

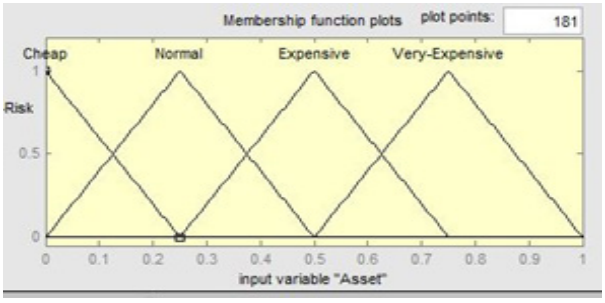



*Fig. 10: Asset Input Variable*

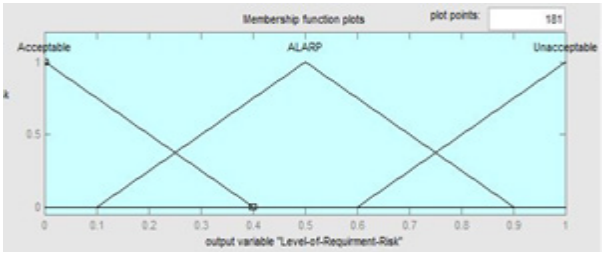



*Fig. 11: Risk Level Output Variable*

### 3- Rule editor (Figure 12):

This is used to add, change or delete rules. It provides opportunity to change the connections and weight applied to the rules (the default weight is always 1).

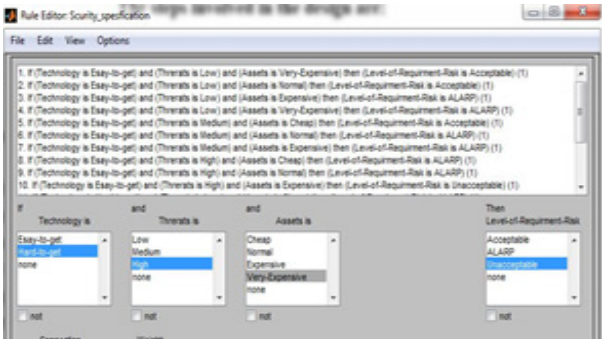



*Fig. 12: Rule Editor*

**4- Rule Viewer (Figure 13):**

The rule viewer shows a graphical representation of each of the variables through all the rules, a representation of the combination of the rules, and a representation of the output from the defuzzification. It also shows the crisp value output of the system. Data are entered for analysis through the Rule Viewer at the Input text field.

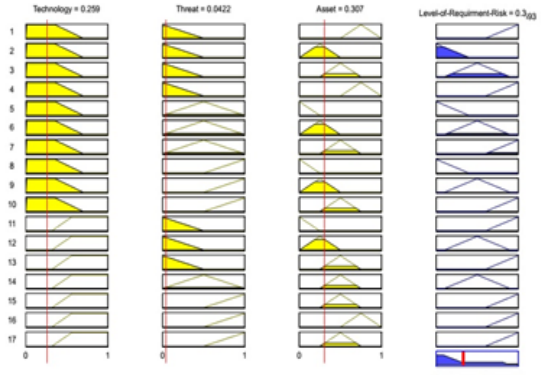



*Fig. 13: Rule Viewer*

As shown the method gives us the level of risk in the previous figure. The next figure, 13 shows the relation and the effect of assets with threat on the level of risk

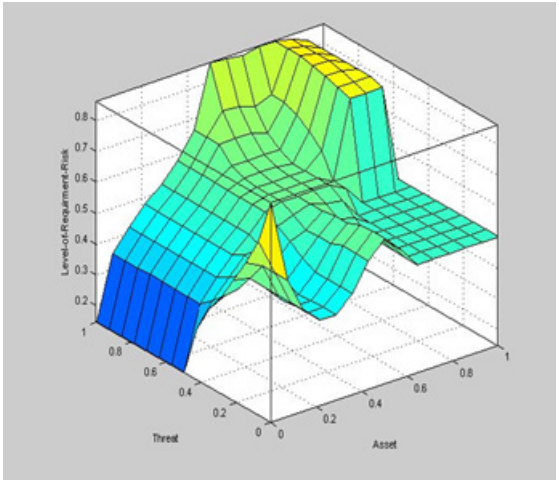



*Fig. 14: Asset & Threat*

The relation between asset and technology is also shown in figure 14

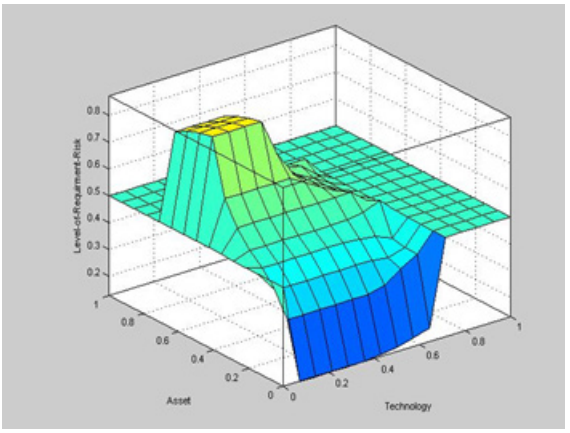



*Fig. 15: Asset &Technology*

## 6. Evaluation

In this section we will test the methodology and find out the results of experiment. The parameters of the input can be adapted by moving the value of any input variable in the rule viewer.

**First Scenario:**

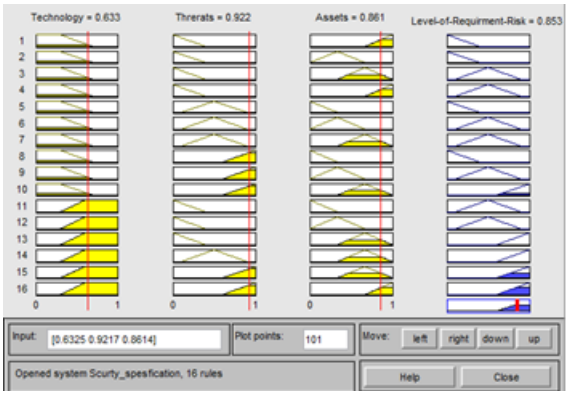



*Fig. 16: Second Scenario Result*

As we see in Figure 16. When the Threat of the security requirement was high with 0.922 %, assets of the organization were very expensive with 0.861 % and Technology was hard to get with 0.633 %. The model predicted that requirement have higher risks with 0.853 %. So the analysis should consider this requirement and find a suitable solution

**Second Scenario:**

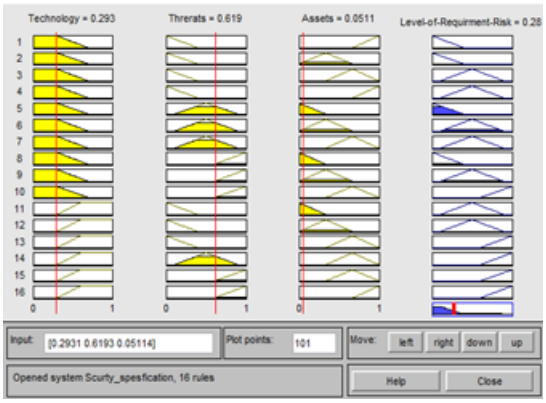



*Fig. 17: Second Scenario Result*

As we see in Figure 17, when the threat of security requirement was medium with 0.619 %, assets of the organization were very cheap with 0.0511 % and technology was easy to get with 0.293 %. The model predicted that the requirement has a higher risk with 0.28 %. So the risk of this requirement can be acceptable and the analysis can devolve it with a confidence of 78%.

**Third Scenario:**

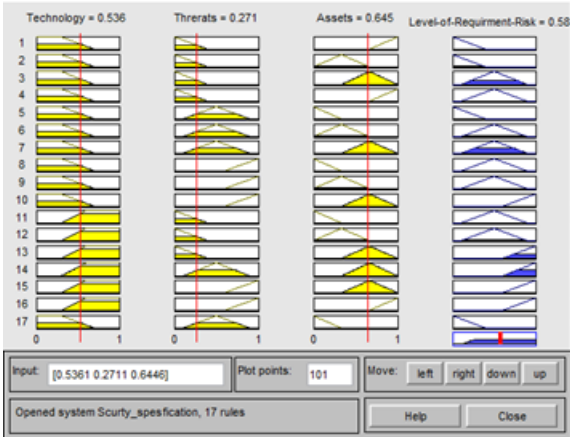



*Fig. 18: Third Scenario Result*

As shown in Figure 18, when the threat of the security requirement was Low with 0.271 %, assets of the organization were very expensive with 0.645 % and technology was easy to get with 0.536 %. The model predicted that the requirement has a higher risk with 0.58 %. So the risk of this requirement can be acceptable so the analysis can devolve it with a confidence of 42%.

As a result, these tools can be helpful for project team analysis that can support and describe the degree of any requirement after adding all rules to the database of those rules.

## 7. Conclusion

In this work, a fuzzy based system was designed to evaluate the possibility of applying security requirement throw specification phase, because it is impossible to provide assurance for the system and justify security measures incorporated unless the system is analyzed during the designing state of computer based systems. With this system designed, risk analysis has been made easier to estimate.

## 8. Future work

For further research, this system is enhanced by redesigning the methodology to dedicate the value of risk for given requirements and then suppose appropriate solutions to mitigate the possible risk.

# References

[1] Sommerville, I.(2007). Software Engineering .BCS/IEE Software Eng. J., 11 (1), 5–18.

[2] Pfleeger, C. P. (1997). Security in Computing, 2nd edn. Englewood Cliffs, NJ: Prentice Hall.

[3] Anderson, R. (2001). Security Engineering: A Guide to Building Dependable Distributed Systems. Chichester: John Wiley & Sons

[4] Ambler, W. S (2005). Introduction to security threat modeling. Agile Modeling. Available at http://www.agilemodeling.com/artifacts/securityThreatModel.htm

[5] Davis, N., Howard, M., Humphrey, W., McGraw, G., Redwine Jr., S. T., Zibulski, G., & Graettinger, C. (2004). Processes to produce secure software: Towards more secure software. A report at National Cyber Security Summit, Vol. 1. Available at http://www.cigital.com/papers/download/secure_software_process.pdf

[6] Sodiya, A.S, Onashoga, S.A, &Ajayi, O.B (2006). Toward Building Secure Software Products. Journal of issues in Informing Science and Information Technology, 3,635-646. Available at http://informingscience.org/proceedings/

InSITE2006/IISITSodi143.pdf Swiderski, F. & Snyder, W. (2004). Threat modeling. Microsoft Press Professional Book Series

[7] Gollmann, D. (1999). Computer Security. Chichester: John Wiley & Sons.

[8] Zadeh L.A., Fu K.-S., Tanaka K. (2014), Fuzzy sets and their applications to cognitive and decision pro¬cesses.Proceedings of the US–Japan Seminar on Fuzzy Sets and TheirApplications. Academic Press University of California, Berkeley, Califor¬nia, July 1-4.

[9] Ross T.J. (2013), Fuzzy logic with engineering applications. Wiley, Vol. 761.

[10] Toval, Ambrosio, et al. "Requirements reuse for improving information systems security: a practitioner's approach." Requirements Engineering 6.4 (2002): 205-219.

[11] Mead, Nancy R., and Ted Stehney.(2005) Security quality requirements engineering (SQUARE) methodology. Vol. 30. No. 4. ACM.

[12] Zimmermann, H-J.(1996) "Fuzzy Control." Fuzzy Set Theory—and Its Applications. Springer Netherlands, 203-240.

[13] Dubois, Didier, and Henri Prade (2012), eds. Fundamentals of fuzzy sets. Vol. 7. Springer Science & Business Media.

[14] Ali, Mohd Hasan, Toshiaki Murata, and Junji Tamura (2004). "The effect of temperature rise of the fuzzy logic-controlled braking resistors on transient stability." IEEE Transactions on Power Systems 19.2 : 1085-1095.

[15] Zadeh, Lotfi A. (1998) "Some reflections on soft computing, granular computing and their roles in the conception, design and utilization of information/intelligent systems." Soft Computing-A fusion of foundations, methodologies and applications 2.1 : 23-25.

[16] Reena Dadhich B.M.(2012), Measuring reliability of an aspect oriented software using fuzzy logic approach. International Journal of Engineering and Advanced Technology (IJEAT).