

Development of a Framework for Applying ASYCUDA System with N-Tier Application Architecture

Ahmad Pahlavan Tafti¹, Safoura Janosepah¹, Nasser Modiri²,
Abdolrahman Mohammadi Noudeh³, and Hadi Alizadeh⁴

¹ Computer Department, Majlesi Branch, Islamic Azad University, Isfahan, Iran

² Computer Department, Zanzan Branch, Islamic Azad University, Zanzan, Iran

³ Science and Research, Islamic Azad University, Tehran, Iran

⁴ Khayam University, Mashad, Iran

ahmad.pahlavantafti@poug.org

Abstract. N-Tier architecture considered as a comprehensive and integrated solution for designing, creating, developing and maintaining the large scale applications. Each application is a set of several physical and logical components. If we want an enterprise application, we should design these components as a well formed layout. N-Tier application architecture provides a model for developers to create a flexible and reusable application. By breaking up an application into tiers, developers only have to modify or add a specific layer, rather than have to rewrite the entire application over. There should be a presentation tier, a business or data access tier, and a data tier. [1]

Using N-Tier architecture would improve flexibility, reliability and extensibility software applications.

ASYCUDA¹ is the client/server application which performs customs declaration and clearance in some countries. ASYCUDA should assist Customs Administrations' modernization and reforms, by supporting both facilitation of legitimate trade and efficiency of Customs clearance controls [2]. It has implemented as a national project in Islamic Republic of IRAN Customs administration since 14 years ago. Nowadays, more than 90% of IRAN Customs processes are done with this system [11].

In this paper we analyze and adapt ASYCUDA application with the N-Tier architecture. We present the integration and validation of ASYCUDA application's tier with the N-Tier architecture and we propose a framework for applying it with the N-Tier application architecture. Then we evaluate some results about our proposed framework and finally we found that, it will make ASYCUDA more flexible and scalable.

Keywords: ASYCUDA, N-Tier Architecture, Customs Application, Scalability, Availability, Integrity.

¹The Automated System for Customs Data is a computerized system designed by the UNCTAD to administer a country's customs.[2].

1 Introduction

Software application is a product which design and create by the software engineers. The processes of design and create a software application are depend on the architecture which they used. Wherever we face to high complexity and large size application, it is necessary to use special architecture. All of the software architectures include the structure of components, relations between components and principles guidance which manage the design and develop of a system [3].

N-Tier architecture is a contemporary model in software engineering. Whenever we use N-Tier model, the modification of the application is easier than other monolithic tier. In N-Tier architecture, the entire core of an application divides into some parts. When we break up an application into tiers, programmers and developers could alter or add a specific layer, rather than the entire application. An application is a set of some parts. Each application must provide ability to store and access data in a consistent manner with a well formed model and it must use a standard graphical user interface and functional processes. A tier is a reusable part of code in an application which performs a specific task. With N-Tier architecture we achieve a flexible and scalable application in a cost effective way [4]. These advantages of this model are caused to choose N-Tier architecture as a goal of this article, indeed.

ASYCUDA is an automated system for customs data and it is a successful application in the filed of the optimizing customs duties.

In this paper, we investigate on the corresponding of ASYCUDA system and N-Tier architecture. The reminder of this paper arranged as follows. The principles of the N-Tier architecture are described in the next section. Comparing ASYCUDA with three types of N-Tier architectures is described in section 3. Proposed framework for applying ASYCUDA with N-Tier architecture is shown in section 4 and evaluation of this framework is described in 5th section. Section 6 illustrates the results and conclusion and the future work are addressed in section 7.

2 N-Tier Architecture

In software engineering, N-Tier architecture is a client-server model in which the presentation, the application processing and the data management are separate logical processes. The most widespread types of N-Tier architecture are the three, four and five-tier architecture.

2.1 Three-Tier Architecture

It was developed by John J. Donovan in Open Environment Corporation (OEC) and it is intended to allow any of the three tiers to be upgraded or replaced independently as requirements or technology change [1]. It has the following tiers:

- **Presentation Tier**, This is the top level of the application and interacts with the users. It means that the end users only work with this tier. It communicates with other tiers via the SOAP, XML and RPC protocols and technologies [5].

- **Business Tier**, It is the logic core of an application. It controls an application's functionalities and responsibilities. All of the application's processes surrounded with this tier. EJB, .NET, CORBA, COM+ and DCOM are the famous technologies and platforms which are used in business tier [6].
- **Data Tier**, This tier performs data storage and retrieval. It is a data repository like the databases or the file systems [5]. Fig.1 shows three-tier architecture.

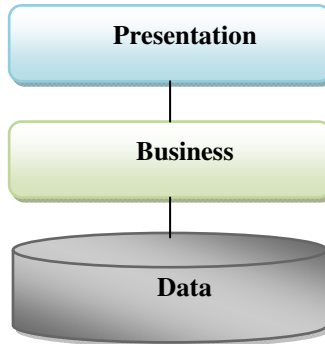


Fig. 1. Three-Tier Architecture

2.2 Four-Tier Architecture

Four-tier architecture is based on the three-tier architecture and pushes one tier between the business and data tier, as a data access tier. The data access tier is responsible for managing and controlling the data tier [7]. You can see this architecture in Fig.2.

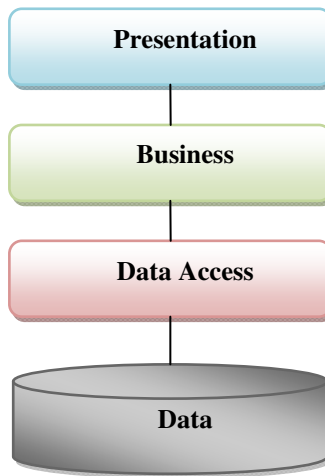


Fig. 2. Four-Tier Architecture

2.3 Five-Tier Architecture

As you can see in Fig.3, this architecture is as same as the four-tier architecture, but it adds a tier on the topmost level. This tier is visualization tier and formats the layouts of an application [8]. Personal computers are the common interface which users work with them. Nowadays, we encounter with other devices like mobile, PDA and so to interact with an application and we accept the differences between these devices and personal computers. Visualization tier is suitable for this purpose. It is physically established on the client machine.

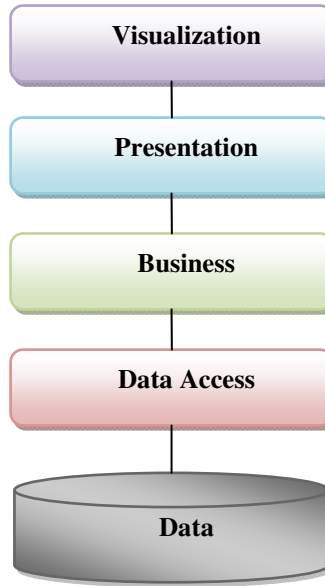


Fig. 3. Five-Tier Architecture

It is better to disturb each tier separately in N-Tier architecture. This architecture has invaluable properties like scalability and flexibility. Distributing each tier on a separate machine could increase the performance and decrease the response time, especially when we have many concurrent users. N-Tier architecture is based on some rules. Each tier could communicate with only its top or bottom tier and could replace with the different tools and technologies.

3 Applying ASYCUDA with N-Tier Architecture

In this section we investigate on applying ASYCUDA system with N-Tier architecture. As we mentioned earlier, ASYCUDA is an application that performs main customs functionalities. ASYCUDA is installed at the request of developing country governments with the assistance of UNCTAD experts [2].

ASYCUDA system migrated customs offices to a modernized customs. ASYCUDA has participated for many years in various working groups and organizations for normalization and it is particularly involved with the WCO working group on the Custom Data Model that has now become simply Data Model after its opening to Other Governmental Agencies (OGAs) [2].

The main objective of the WCO Data Model is to define a set of standardized Data usable by both Customs Administration and trade operators, for electronic data exchange within the Customs Clearance process (manifest, declarations ...) [2].

Is ASYCUDA corresponded with N-Tier architecture? In Table 1, you can see the true and the false alerts of corresponding between the ASYCUDA system and N-Tier architecture. As you see, ASYCUDA is not applied with five-tier architecture.

Table 1. The true and the false alerts for applying ASYCUDA with N-Tier Architecture

| Applying ASYCUDA with N-Tier Architecture | | | | | |
|---|-------|-------------|----------|--------------|---------------|
| Architectures | Tiers | | | | |
| Three-Tier | Data | Business | | Presentation | |
| | TRUE | TRUE | | TRUE | |
| Four-Tier | Data | Data Access | Business | Presentation | |
| | TRUE | TRUE | TRUE | TRUE | |
| Five-Tier | Data | Data Access | Business | Presentation | Visualization |
| | TRUE | TRUE | TRUE | TRUE | FALSE |

4 Proposed Framework

Table 1 shows the compatibility between the N-Tier architecture and ASYCUDA system. According to the false alert in visualization tier, between ASYCUDA and five-Tier architecture, we attempt to find architecture to improve quality of the proposed framework. In this framework, we add an abstract tier for visualization the user interface. All of these tiers are separated in the framework. Fig.4 illustrates the model of the framework.

Visualization tier in our proposed framework aims to reformat the user's request and response and permits the users could communicate with ASYCUDA with any computer devices like PDA, Personal Computers, Laptops and etc. the WAP and IMAP4 usually reformat the user interface for PDA and Mobiles computers. XSLT and XPATH are a language that describes how to locate specific elements (and attributes, processing instructions, etc.) in a user documents. It allows users to locate specific content within an XML document [9].

Recommendation framework attempts to show a set of modern technologies with the highest degree of abstraction and connectivity between tiers.

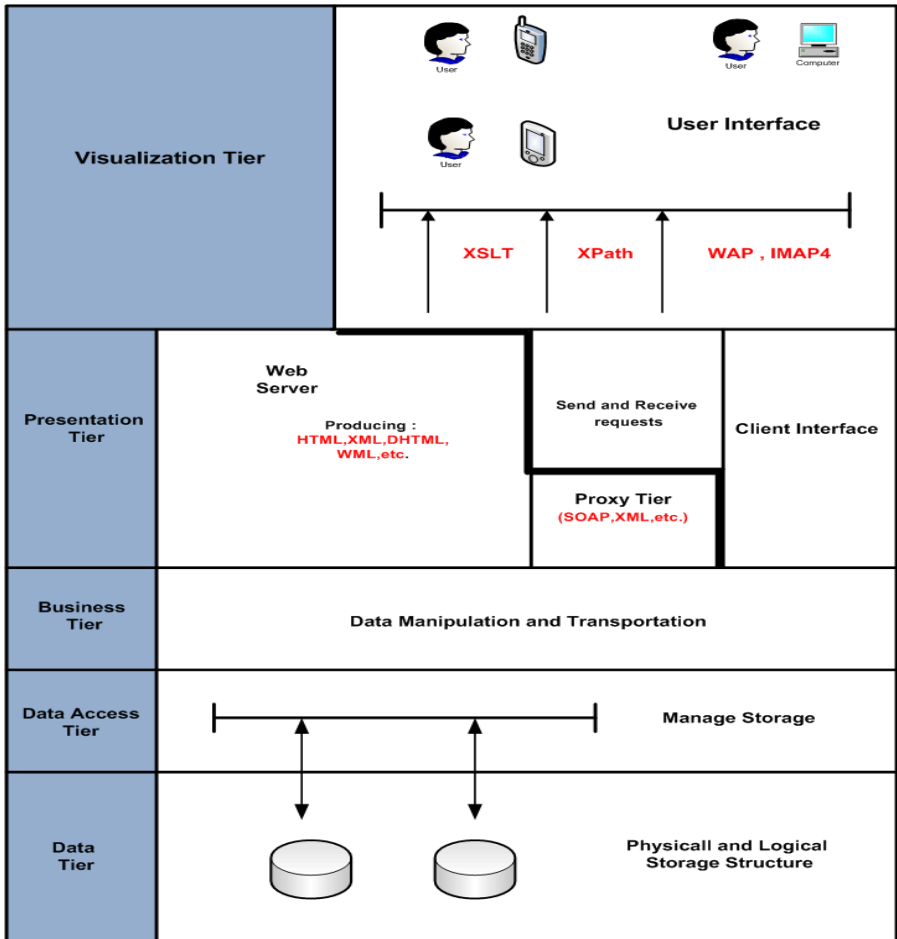


Fig. 4. Proposed Five-Tier Framework for ASYCUDA

Proxy is an object or person authorized to act for another [10]. In our framework, Proxy is referring to any sort of code that is performing the actions for clients. The client interface is connected directly to the business tier.

The topmost level of this architecture incorporates the user interface to present output information and also to route the input parameters as well as user actions to the presentation Tier.

5 Proposed Framework Evaluation

Different types of software properties are used to evaluate the capabilities of the recommendation framework. We listed several software architectural properties in order to evaluate the characteristics of our proposed framework. The software architectural priorities which we have classified are Flexibility, Openness, Modularity

and Scalability. We choose 50 persons of internal ASYCUDA users, 50 persons of external ASYCUDA users and 50 persons of the ASYCUDA domain experts and 50 persons of software architecture domain experts. We choose Khorasan (Mashhad) customs Administration office for our dataset. It is a branch of Islamic republic Of Iran Customs Administration (IRICA).

Opinions of the Internal and external users of ASYCUDA system are shown in Figure 5.

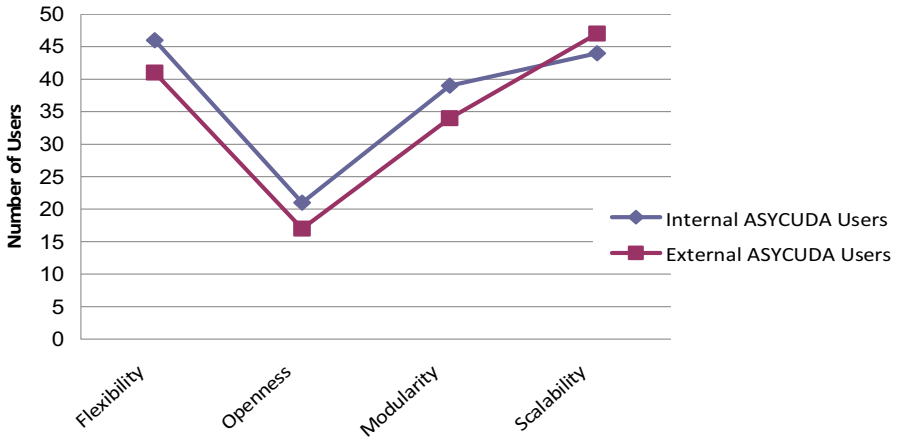


Fig. 5. Internal and External Customs User's Viewpoints

Figure 6 shows the evaluation of the domain experts.

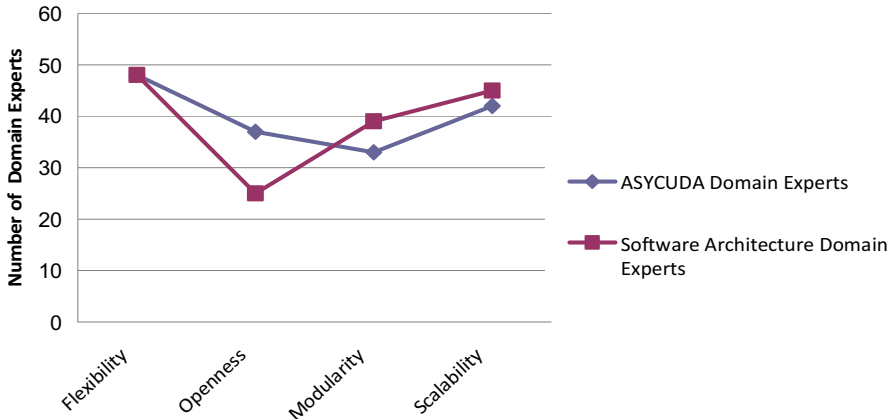


Fig. 6. Evaluation of the domain experts

6 Results

Figure 5 and Figure 6 demonstrate the results of the polls which have done at Khorasan Customs. As you see, these diagrams display some fluctuations percentage of some properties that implemented by the proposed framework.

Figure 5 illustrates that our framework is more scalable in external users' opinions, about 95%, while it is approximately 80% for internal users. Also this diagram demonstrates some parameters such as openness, modularity and scalability which the external users believe that it is more scalable, approximately 95% whereas it is about 88% for internal users. The percentage of openness and modularity is lower than the others. For instance, results of the evaluation show that the openness of our framework is less. Also percentage of modularity in proposed framework is about 78% for internal users and 70% for external users.

Figure 6 displays variety of many experts' viewpoints about percentage of some architecture's characteristics. According to our evaluation, a large number of experts almost 98% believe that the system is flexible and more than 85% of them believe that our proposed framework is scalable, but the system is not openness enough almost 50%. The people who have some experiences in ASYCUDA system and customs domain believe that the system has less modularity, approximately 68%.

Our proposed framework makes more flexibility for ASYCUDA application in a scalable manner.

7 Conclusion and Future Work

The software applications of customs must be as comprehensive, open and integrated tools which are able to address the needs of expected business.

To design and create a more efficient customs application that will facilitate trade, we represent an integrated solution for applying ASYCUDA system with a flexible and scalable architecture.

Multi-Tier architecture has several benefits such as integrity, availability, flexibility and scalability. In N-Tier model, every layer is independent from others; it means we could modify each layer easily, without changing any thing in other layers.

In this paper, we used the advantages of N-Tier architecture and improve ASYCUDA architecture as a customs digital system. We design a framework as Five-Tier architecture to increase the ASYCUDA capabilities in scalable and flexible manners.

In the future we will study about new generation of customs applications and we will represent a software architecture which is suitable for modern customs.

References

- [1] <http://en.wikipedia.org>
- [2] <http://www.asycuda.org>
- [3] Edwards, J., Devoe, D.: 3-Tier Client/Server at Work. John Wiley & Sons, Inc., Chichester (1997)
- [4] Tanenbaum, A., Steen, M.V.: Distributed Systems, 2nd edn., Prentice-Hall, Englewood Cliffs (2007)
- [5] http://en.wikipedia.7val.com/wiki/multitier_architecture

- [6] <http://publib.boulder.ibm.com/infocenter/wasinfo/v5r0/topic/com.ibm.websphere.exp.doc/info/>
- [7] Mckinely, P., Sadjadi, S., Kasten, E., Cheng, B.: Composing Adaptive Software, January 2004. IEEE Computer Society Press, Los Alamitos (2004)
- [8] http://en.wikipedia.org/wiki/visualization_computer_graphics
- [9] Nanda, M., Karnik, N.: Synchronization Analysis for Decentralizing Composite Web Services. International Journal of Cooperative Information Systems 13(1) (2004)
- [10] <http://www.m-w.com/cgi-bin/dictionary?proxy>
- [11] <http://www.irica.gov.ir>