

Implementation of Robust and Effortless Multi-Tenant SaaS Framework at Single Instance

Apurva Sonawane

Department of Computer Engineering
AISSMS College Of Engineering
Pune, Maharashtra, India

Ashwini Dhekale

Department of Computer Engineering
AISSMS College Of Engineering
Pune, Maharashtra, India

Hoda Hossein

Department of Computer Engineering
AISSMS College Of Engineering
Pune, Maharashtra, India

Priyanka Kakade

Department of Computer Engineering
AISSMS College Of Engineering
Pune, Maharashtra, India

N. R. Talhar

Department of Computer Engineering
AISSMS College Of Engineering
Pune, Maharashtra, India

Abstract

Multi-tenancy is most important concept of SaaS. Multi-tenant SaaS model provides license to the client for accessing the services. It provides services to thousands of clients using single web tenant. Software as a service is used by the clients as a software service.

This paper describes effortless multi-tenancy at a single instance with the help of multi-tenant SaaS framework. By using dot net technology this framework is being implemented with various features. Effortlessly we can add n number of databases to assign to specific tenant. Scheduled tasks is included in multitenant SaaS framework which performs background operations and helps in optimizing the portal. Various features like bulk insert, usage metering, tenant encapsulation, event logging and mainly effortless multi-tenancy is also provided by this robust framework.

Keywords— multi-tenant, SaaS, SaaS Framework single instance, effortless multi-tenant.

I. INTRODUCTION

Multi-tenant SaaS framework architecture is implemented with various technology. Asp.net technology is being used for implementation. The goal is to provide and manage portability, security, scalability and customization. In this paper our aim is to implement multi-tenancy at single instance. In this paper we have introduced the SaaS framework for different windows applications and web.

SaaS platform can be considered as type of specialized app server like IIS server in this

architecture. Different Asp.net website are included in IIS server and then another website is created in IIS with the help of portal wizard with the help of different IP and DNS. But each website will go under different work processes and different application pool. So both having different work process and same instance. We are creating cloud like environment it does not include actual private or public cloud. In physical manner mainly we are using the computing resources [1]. Various features like bulk insert, usage metering, tenant encapsulation, event logging and mainly effortless multi-tenancy is also provided by this robust framework.

II. OBJECTIVE

Single instance multi-tenant is an architecture which provides single instance of software application that serves multiple clients. Pre-defined set of pages, components etc are contained in template which would be automatically created for tenant when one of the available templates is chosen by clients. Third party can easily integrate and collaborate with external resources. We can easily add extra modules without interrupting other module. The older version of the tenant is replaced with the newer version of the same tenant which is also known as easy framework upgrade. One of the main objectives of this framework is effort less multi-tenancy.

III. EXISTING SYSTEM

In the earlier systems, the concept of multi-tenancy was defined and benefits of the same was explained and also why metadata driven architectures were the premier choice for implementing multi-tenancy was

demonstrated. Like other systems this system also had its benefits and lacks. The system lacked in providing any detailed information about the event log, no information was provided about the user profile, no mapping was provided and also no algorithm was provided for the customer account security.

IV. PROPOSED SYSTEM

Problem definition of proposed system includes:

1. Implementation of multitenant SaaS Framework at single instance and with effortless multi-tenancy.
2. Implementation of RSA algorithm for security to proposed framework.
3. Providing different features to framework.
4. Providing services according to
 - a) Presentation layer
 - b) Business layer
 - c) Data access layer

As, all existing vendors provides multi-tenancy at multiple instances. So, it is very time consuming and hectic to create multiple instances and also load balancing should be provided for different instances. To avoid time consumption and other resources use in proposed system framework is created so that, multi-tenancy will be provided to different tenants at single instance. Also, effortless multi-tenancy is provided to reduce effort to give services to customer.

V. PROBLEM DEFINATION

The overall objective of the project is to overcome different challenges and problem definition consist of:

1. To implement SaaS in Cloud Computing [2]
2. Service Job Scheduling System for Cloud Computing [3]
3. To provide services according to layers
 - Presentation Layer
 - ▶ web Forms
 - ▶ Skins
 - ▶ Module user controls
 - Business Layer
 - ▶ Localization
 - ▶ Caching
 - ▶ Exception Handling
 - ▶ Event logging
 - ▶ Search
 - ▶ Installation & Upgrades
 - ▶ Security

- Data Access Layer
 - ▶ User Accounts
 - ▶ Log Access
 - ▶ Digital Signature with RSA algorithm for Secure data in database[4]
 - ▶ To control User data and information by MD5 [5]

VI. MODULE DESCRIPTION

The modules that are included in this framework are as follows:

- Creation of template
- SaaS architecture
- Integration of template and SaaS architecture
- Services used by the admin

1. Creation of template

- Template contains a pre-defined set of pages, components, services etc.
- It would be automatically created for tenant when they choose one of the available templates like small-scale organization, mid-scale organization, large-level organization.

2. SaaS architecture

- The Multitenant SaaS architecture will include different servers to serve the request.
- The web server will contain different layers like data access layer, BI layer and finally presentation layer.

3. Integration of template and SaaS architecture

- It puts the SaaS multi-tenancy operation in the complete life cycle of Cloud platform services.
- It describes the method of creating multi-tenancy enabled virtual images, creating the application model, transforming the application model into topology model, instantiating the virtual images, then activating the image and making additional installation and configuration including the multi-tenancy related actions.

4. Services used by the admin

- SaaS administrator creates the web portal for every tenant and creates the database for that web portal at single instance and with zero effort multi-tenancy.
- SaaS administrator will have authority to create web portal and database for that web portal.
- Then that tenant will rent for that service and license is created. Once validly expires, that license will be renewed.

VII. SYSTEM IMPLEMENTATION

We are going to take the help of Dot Net technology to implement this framework. After implementation of this multi-tenant SaaS framework at single instance and effortless multi-tenancy, we come to a result where different websites are created to serve different clients.

Here, mainly three steps are followed in order to complete a SAML post, they are as follows:

1. Based on XML standard an XML transaction will be build.
2. Later we have to properly sign the XML.
3. And lastly Base64 encoding is done.

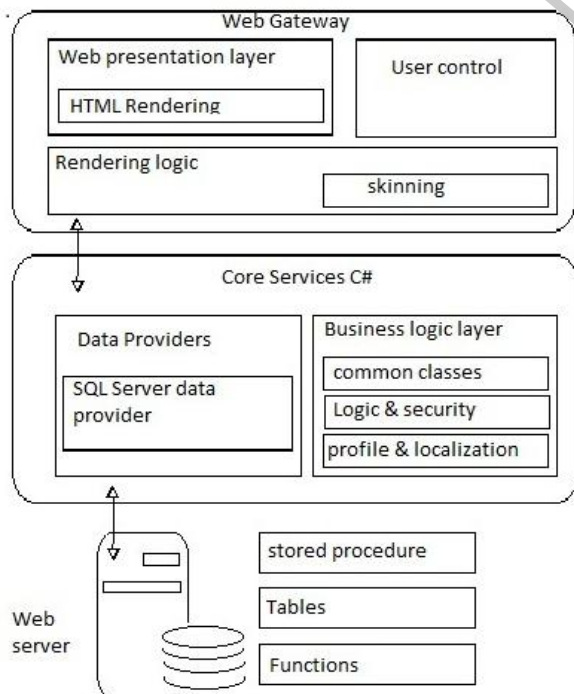


Fig.1 Proposed multi-tenant SaaS Architecture

URL generation and database creation of website at single instance is the main part in the implementation of this framework. At runtime a mechanism for mapping virtual resource names to physical resource names is provided by the URL rewriter which is an HTTP module. For tenant access a friendly URL is provided. The vendor provides services to the tenant on pay basis which can be either paying fees monthly or annually.

In this model, software can be accessed from anywhere over the internet by the software users. SaaS services can be used by the tenants through web browsers. The software web portals are implemented after which separate databases are created.

After giving the license to the client, invoice will be generated according to the features and templates provided to the web portal and the client will be given the deadline for payment .

Basically when the tenant is created , an account for every client is created and various features like event logging, dashboard , metering facility will be provided to the client.

VIII. CONCLUSION

In this paper, we explore and then provide the implementation of multi-tenant SaaS framework with effortless multi-tenancy and at single instance. Here we conclude that tenant gets user friendly and efficient web application by using the features of this framework. This website is provided to the tenants on rent basis by getting license of particular period of time. As this framework creates cloud like framework there is no need to create cloud. Multi-tenant SaaS provides service to thousands of clients using single web tenant. Software as a service is used by the clients as a software service. This framework creates websites in very less time which is very useful for the developers. Various features like bulk insert, usage metering, effortless multi-tenancy, tenant encapsulation, event logging is also provided by this robust framework. In future multi-tenant SaaS framework would support mobile browser.

ACKNOWLEDGEMENT

This paper involves number of respected helping hands. We are grateful to Prof. N. R. Talhar for his dedication and valuable guidance. We would like to thank the Department of Computer Engineering, AISSMS COE, Pune for their uninterrupted help and support.

REFERENCES

- [1] Jihyun Lee Sung Jin Hur ,”*Level 2 SaaS platform and platform management framework*” Advanced Communication Technology (ICACT), 2011 13th International Conference.
- [2] Manish Godse, ShrikantMulik, "An Approach for Selecting Software-as-a-Service (SaaS) Product," cloud, pp.155-158, 2009 IEEE International Conference on Cloud Computing, 2010
- [3] Thomas Erl, SOA Design Patterns, Prentice Hall PTR, Upper Saddle River, NJ, 2010
- [4] Brian Moore, Qusay H. Mahmoud, "A service broker and business model for saas applications," aiccsa, pp.322-329, 2009 IEEE/ACS International Conference on Computer Systems and Applications, 2010
- [5] M.T. Hoogvliet, “SaaS Interface Design-Designing web-based software for business purposes,” unpublished.
- [6] Hai Henry, Sakoda, Seitaro, “SaaS and integration best practices”, F David Banks, John S. Erickson, Michael Rhodes, “Multi-tenancy in Cloud-based Collaboration Services”, Hewlett-Packard Development Company, L.P., February 21, 2009 Scientific and Technical Journal, v 45, n 3, p. 257-264, July 2010
- [7] Paul Giurata, “How to Use Customer Life Cycle as a Strategy for SaaS,” unpublished. V4
- [8] David Banks, John S. Erickson, Michael Rhodes, “Multi-tenancy in Cloud-based Collaboration Services”, Hewlett-Packard Development Company, L.P., February 21, 2009
- [9] Jinan Fiaidhi, Irena Bojanova, JiaZhang ,Liang -JieZhang , “Enforcing Multitenancy for Cloud Computing Environments” January/February 2012 (Vol. 14, No. 1) pp. 16-18 1520-9202/12/\$31.00 © 2012 IEEE
- [10] Sungjoo Kang, Sungwon Kang, SungjinHur, ”*A Design of the Conceptual Architecture for a Multitenant SaaS Application*” 2011 First ACIS/JNU International Conference on Computers, Networks, Systems, and Industrial Engineering.