

Proactive Real Time Data Migration/ Synchronization Approach For SAAS Application In Cloud Platforms

K. Abdur Rahman Khan^[1], Dr. P. Suryanarayana Babu^[2]

^[1]Lecturer &, Research Scholar, Department of Computer Science
Osmania College, Kurnool

^[2]Principal, Research Supervisor, Department of Computer Science
Sri Vijayadurga Degree College, Kurnool
, Rayalaseema University, Kurnool
Andhra Pradesh. India

ABSTRACT

Data migration / synchronization present multitudinous demanding for organizations looking to create consistent integration. The current Enterprises applications tend to be heterogeneous environments. They are composed / designed of many various systems, each with their own data-structures and specification. This discrepant structure creates confusion as each end-point calls for its own complex data mappings, type-setting, estimations, and data aggregations /accumulations. Additionally, real-time efficient and reliable data synchronization, a growing trend in SaaS integration, is very difficult to develop with huge data volumes. This paper illustrates the secured and reliable data migrations and synchronization use cases in the real time enterprise applications.

Keywords:- SOAP, REST, ERP, CRM

I. INTRODUCTION

SAAS (software as a service), a customer contracts to use any cloud based application, such as Enterprise Resource Planning (ERP) or Customer Relation Management(CRM), hosted by a third party, rather than purchase a software original license and installing the application software on its own environment. Just as consumers can check e-mail or use mapping programs/ applications with their related Web browsers, so that customers can access their enterprise business applications over the web/Internet [1].A novel enterprise uses various many applications across their premises which are not default designated with out of box feature. Each every applications have various layers namely data, presentation, business logic, and security layers. Security layer integration is one of the key aspects for making application integration across the enterprise. Cloud security can achieve with various development tools (security source code scanner, chimera). Analyzed various data integration patterns[2]. Real-time Data layer integration provides the usage of common shared data and effectively manages data across the multiple applications. Data layers integration can achieve with various methodologies (like bulk data transfer API's, Language specific tool kits (ex: java, java script, Ruby), Cloud-to-Cloud (Google, Facebook), and open API's

(SOAP, REST)).Business Logic often span across the various enterprise application.to implement the complete end-to-end business process functionality we have to enhance the related existed business logic from one system to another. Business logic integration can be achieved with messaging systems, web services, and specific language callouts. Presentation layer integration which helps the organization to access various types of templates, canvas, connect with social media, accessing existed web pages across the multiple applications. Presentation layer integration can achieve with various software development kits (SDK's) [3].

II. ENTERPRISE ARCHITECTURE

Cloud data migration is critical and essential metric for business to have a efficient tools and techniques to do so.when moving from one cloud to another cloud ,it is very important that all of the organizational data migrates without any complications.The following diagram illustrate the secured data integration to achieve the reliable data migration / bi-directional synchronization between cloud and on premises enterprise systems.

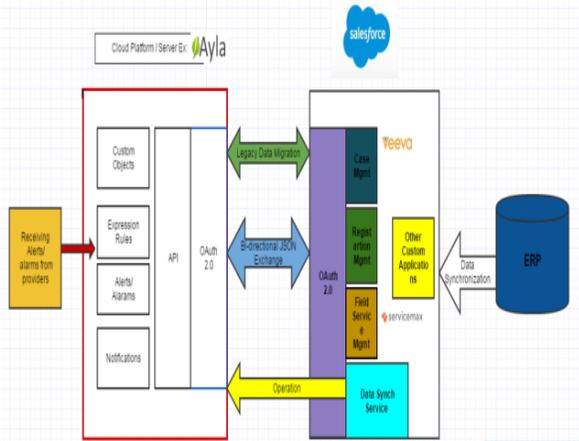


Figure 1: Enterprise Architecture for Data Integration / Migration / Synchronization

III. METHODOLOGY

The information or data will be extracted and updated on Ayla using a Custom Object which will be written for these purposes. Although the output and input will be in CSV file format but may need some formatting, e.g., column names, field widths, etc. to be acceptable on SFDC and vice versa for the CSV extracts generated from SFDC. The data synchronization done with the batch related job, which is scheduled at particular time. This batch will get the all changed data and insert into Ayla web service. The overall process flow should take place in two steps, in the first stage workflow need to configure at SFDC end and second is configure the batch to update the records at Ayla end to get the connection between to Ayla and SFDC, I configured remote site setting in SFDC end. Remote site setting have a details about the end point URL of the destination cloud.

The data synchronization between the two clouds done by calling the rest based web services, which will fire a series of steps at both end of the cloud applications. In the Ayla side the series of process as follows : from SFDC end user must login with valid sandbox credentials, User will get the session id from the authorization services, user send the custom objects details to rest based web service which have a synchronization custom logic. If the web service identify the id of the particular record then it will synchronize otherwise it create a case record due to failure. From the SFDC end the series of operations as follows: User need to configure on workflow rule (Figure:2) along with the outbound message configuration action (Figure:3),

which have an end point URL of the integration interface along with criteria. The criteria are whenever the update the particular records immediately it check whether the monitored fields are changing in the object. If the monitored fields have changed then status of the synchronization updated as pending then it calls the batch job. The batch job divides the huge no of records into smaller chunks based on the synchronization criteria and then calls execute operation.

The execute operation call the web service by making HTTPs REST call out, and gets the Response body either in XML(Extensible markup language) or JSON (Java script object Notation).JSON data is parse with JSON parsers and DE serialize into the SFDC objects. For every record once the DE serializes happened the synchronization status will be updated to complete.

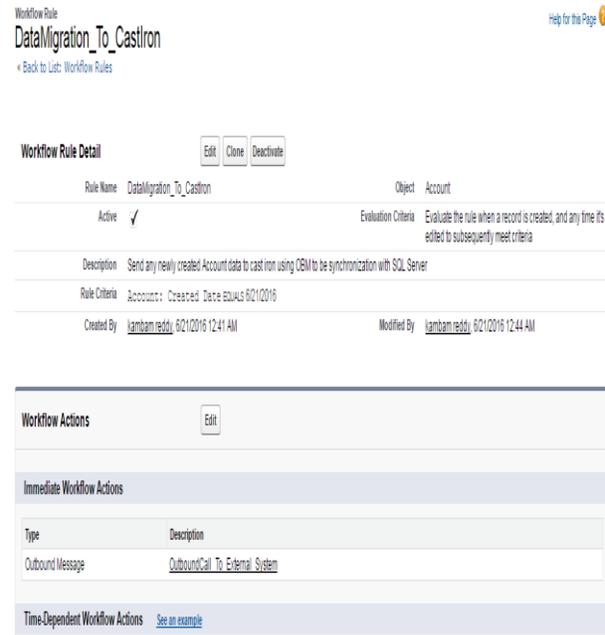


Figure 2: Configuration of workflow rule

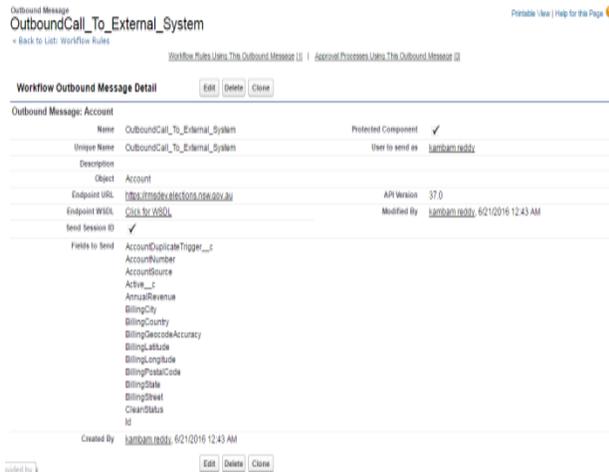


Figure 3: Configuration of outbound message for Work-flow rule

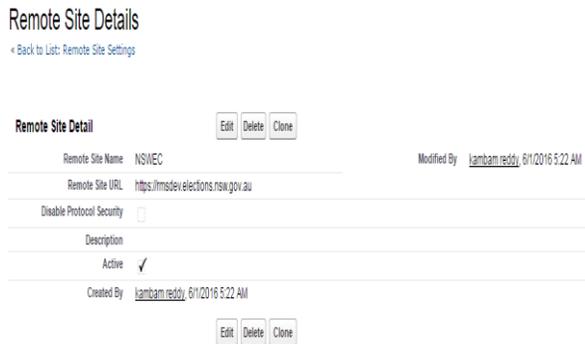


Figure 4: Remote site settings to call external enterprise applications

IV. CONCLUSION

There are many research challenges/target to integrate cloud applications with on-premises which includes, data migration, data fusion, data synchronization, SAAS integration with IoT Cloud platforms, data fetching from on-premises to cloud environments. Reliable SAAS

integration with on-premises always is an asset for organization to unseal the potential value of their Cloud investments.

Our proposed data migration/synchronization strategy provides a unified data synchronization platform that addresses the organization issues with real-time reliable data synchronization and bulk data migration, High data transforms and loading capabilities. Our comprehensive strategy allows organizations, end users to perform reliable data synchronization for SAAS applications. The data synchronization reduces the SAAS investments. The REST based web service strategy is very light weight and addresses the communication between to different platforms.

The proposed methodology reduces the development cost and time of organizations to reach their ROI (Return of Investments)

REFERENCES

- [1] A Dubey and D Wagle , "Delivering software as a service" , The McKinsey Quarterly, Web exclusive , 2007
- [2] <https://www.mulesoft.com/lp/whitepaper/saas/salesforce-integration-patterns>
- [3] https://developer.salesforce.com/docs/atlas.en-us.integration_patterns_and_practices.meta/integration_patterns_and_practices/