Design of a Cost Efficient Subscription Engine to Improve Customer Experience in Value Added Services

Shah Mosaddek Hossain[1], Kingshuk Reza[2], Ahmed Mostahsan[3], K.M.A Salam[4] and Md. Hasan Uz Zaman[5]

Now-a-days value added service providing to mobile phone users is a popular business. Focusing on this, third party Content Providers (CP) offers various kinds of services for telecom subscribers, which are called VAS or CP services. Content providers are providing these services through different telecom channels like SMS, IVR, USSD, WAP, etc. Based on customer feedback, in many cases customers are forcefully subscribed to the services by the content providers and also charged customer for that and most of the operators has no control mechanism to stop this. In this paper we are proposing a solution to capture the value added services in a single subscription engine which will be a low cost solution, efficient, customer centric, full control on subscription mechanism, single GUI solution for customer care agents, reduce customer complaints and will improve customer experiences in VAS domain.

Keywords: Subscription Engine; Content Providers; Short Codes; MPBN; SMS; USSD.

Field of Research: Computer and software engineering.

1. Introduction

A service delivery platform (SDP) is a platform that provides a structure for service delivery, including controls for service sessions and protocols for service use. This term is common within the telecommunications industry. It can be vital for services that have to bridge multiple platforms or technologies. However, a service delivery platform tends to be designed for a particular delivery in a single telecommunications format. The design of a subscription engine is come from the concept of service delivery platform. In Telco market there are commercial off-the-shelf (COTS) products available but most of them are expensive with specific features. This paper depicts how to design a cost efficient subscription engine, the impact of it in a telecom operator's service layer, with a vast customization of features compare to service delivery platform. Subscription engine its act as a control box to ensure end to end service delivery through various channels like SMS, USSD, IVR, WAP etc. Another reason to install subscription engine in a telecom network is to ensure security to its mobile packet backbone network (MPBN). This paper also describes the security system to protect internal service nodes and the control mechanism of a Subscription engine.

_____

[1, 2, 3] PG Students, Department of Electrical and Computer Engineering, North South University, Bangladesh
E-mail: mosaddek2005@yahoo.com
[4, 5] Associate Professor, Department of Electrical and Computer Engineering, North South University, Bangladesh
E-mail: asalam@northsouth.edu

2. Content Providers and Service Modality

Before describing the whole thing, it is important to explain some terminologies, used with respect to Subscription Engine. Following are some terminologies; those will be widely used to describe Subscription Engine:
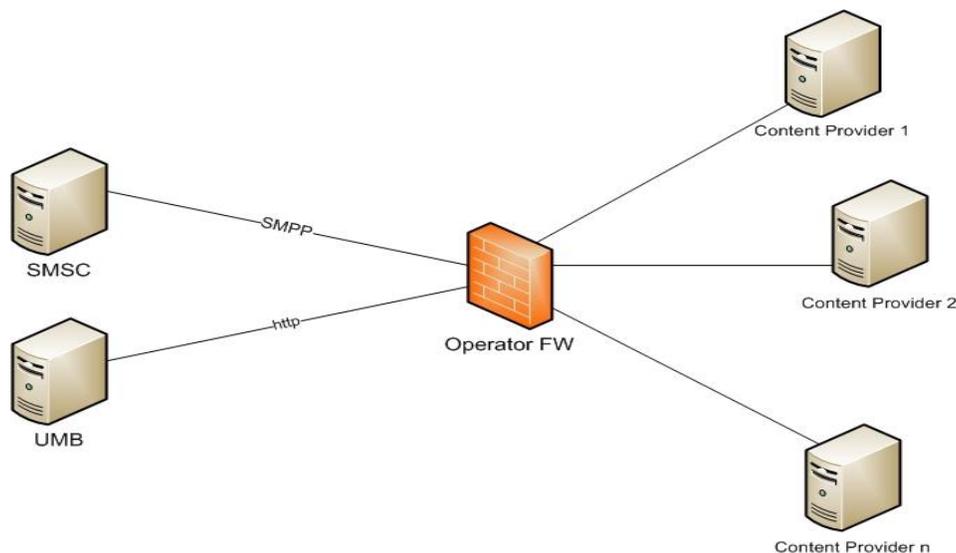
- CP: Content Providers or CPs are third party companies or entities, who will be providing Services to the Subscribers. They will own CP systems, which will be interfacing with Subscription Engine.
- CP Systems: Systems at CP end, which will be communicating with Subscription Engine by pre-defined APIs
- Services: A Service is one specific modality of providing SMS content to subscriber by a CP. Services can be many types, among which two are considered in the scope of this SRS. These two are – SMS broadcast and SMS push-pull.
- SMS broadcast: This is a type of Service, to get which a subscriber needs to register for it. CP will broadcast SMS contents to registered subscribers in a regular basis. Subscribers can de-register from the service any time to stop receiving contents.
- SMS push-pull: This type of Service, to get which a subscriber needs to request for in by SMS in ad-hoc basis. CP will provide the content to subscriber in response of that request.
- Short Codes (SC): SC is an SMS port, through which SMS can be sent or received to or from subscribers.
- Key-word: Pre-defined string value, which need to be sent by subscribers by SMS to register/de-register to/from broadcast service or to get content of push-pull service.

Third party content providers are mainly provides different types of contents like jokes, news, weather report, sports news etc. through SMS channels. They have their own system with different types of control mechanism. In current architecture they reserve the subscribers' database at their end and send mobile terminating SMS based on their own database. Basically content providers provide SMS services using SMS short codes like for example: 16235, 16291etc. Content provides need legal approval from concern telecom authority of the country. From Bangladesh perspective Content provides get these types of short code from Bangladesh Telecommunication Regulatory Commission. SMS short codes contain different types of charging based on service modality. Mainly two types of SMS service are mostly popular push pull based and subscription based. Push pull services are interactive type service, when subscriber send an SMS to a particular short code the customer will receive a SMS with desired information immediately and in subscription based services the customer need to subscribe in a particular service and he/she will get desired content only daily, weekly, monthly basis or as per service modality. Content providers also have WAP, IVR, USSD based services. SMS channels are also used for WAP and USSD notifications in existing systems. (Wikipedia, 2014)

3. Challenges Without A Subscription Engine

In the traditional systems the service nodes (SMSC, UMB etc.) are connected with the third party content providers directly using SMPP protocol or via SMS/USSD middleware systems using web services.

Figure 1: Direct Connectivity with Service Nodes without Using any Middleware Node

The major disadvantages of this kind of architecture are:

- No control mechanism on TPS (transaction per second) throughput
- Third party content providers are easily misuse of this type of direct connectivity as operators don't have any control on their content or on TPS throughput.
- If the system is hacked and then there is a high risk of getting hacked of the entire core/service node which are hosted in the Mobile Packet Backbone Network.
- In efficient architecture as the content providers need to push content through SMS one by one to each subscriber. For example, if a1000 subscribers subscribed for any SMS service then the content provider needs to push 1000 times by using given API of the service provider which is an inefficient process.
- Content providers can easily register or de-registered any subscriber in to any service.
- Content providers can charge customer through terminating SMS.
- Content provider can do any type of campaign without notifying the respective service provider using their assigned SMS short codes.
- No single screen solution available for customer care agents of the respective operator/service provider in this type of architecture.
- As a result increased numbers in customer complain.
- Poor service experience.

4. Introducing Subscription Engine with its feature

In this paper an improved architecture of security system has introduced with the subscription engine. The Subscription Engine will be a standalone web service, which will have interfaces with SMSC, Content Providers' system (CP systems), CRM system and Revenue Assurance system (RA). Some other systems like UMB will also be able to send request to Subscription Engine. (The SDP Alliance, 2006)

Through SMSC, Subscription Engine will send and receive SMS to and from subscribers. Send SMS can be triggered in various circumstances like broadcasting SMS contents provided by CP, push-pull response, registration/de-registration response, etc. Subscribers can register/de-register by SMS or by UMB menu. CRM interfaces will also have some features to view subscribers' information and provisioning of Services. DWH (data ware house) and Revenue Assurance will be fed with data from Subscription Engine to prepare reports and set control points over the system.

In Subscription Engine, System Administrators can configure CPs, SCs and Services. For active Services, subscribers can register/de-register for broadcast type service or request for content in case of a push-pull type service. These requests will be handled by the Subscription Engine and replied to subscribers accordingly.

When a subscriber sends an SMS to a SC, Subscription Engine will receive the SMS and validate the key-word to determine the type of request. The request can be either a registration/de-registration request to/from a broadcast service or a content request for a push-pull type service. For registration/de-registration request, Subscription Engine will perform requested activity, notify CP about it and reply back to the subscriber. For push-pull request, it will get the content from relevant CP and send it to the subscriber.

For broadcast type services, CP system can request Subscription Engine to send content to all subscribers of a particular Service. Subscription Engine will than validate the request and broadcast the content to all subscribers of that particular Service. (Net Cracker, 2014)

A subscriber can also call customer service or come to customer points with complains / requests / clarifications. To facilitate customer Engines to handle queries, Subscription Engine will also expose necessary APIs to CRM system. CRM system can check subscribers' information, service information, etc. and can register/de-register a subscriber to/from a broadcast type Service. The same set of APIs can also be used by other systems like UMB, IVR, etc. to perform these activities. Subscription Engine will keep track of source system, which are using this set of APIs. (Metaswitch Networks, 2011)
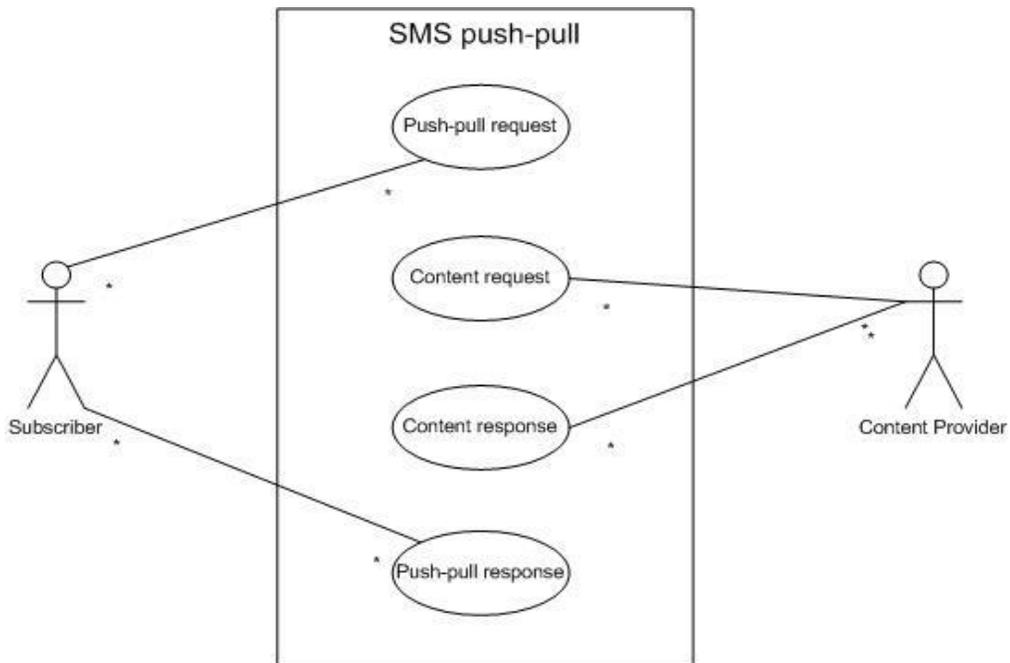
Subscription Engine will keep track of all transactions along with necessary data. Relevant data will be fed to DWH and RA, by which they will generate reports and set control points in the system. (Huawei, 2014)

Figure 2: How Communication Will Take Place in the New System
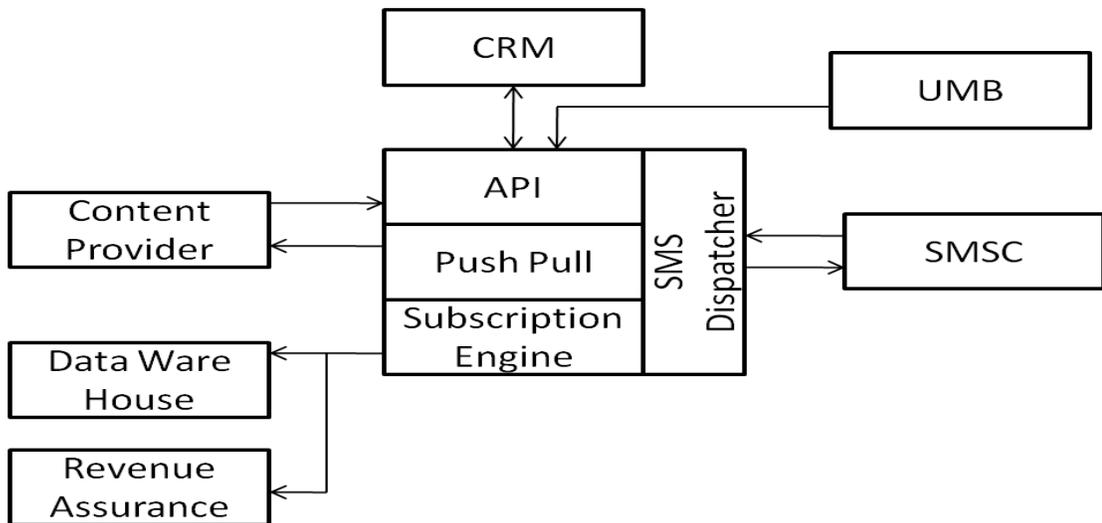
Push-Pull SMS Mechanism

**Figure 3: How Push Pull Communication Happens**

Through this feature subscribers can request for content by sending a pre-defined SMS text to a specific port and in reply SMS he/she will get content. For example, there can be a service exposed to a specific port for cricket score. If subscriber sends a pre-defined keyword to that port, in reply he/she will get score by SMS.

- The system will be able to identify whether the subscriber request for registration or push pull content requests (fetch from CP end) depending on pre-defined message format. This message format will be defined during Service configuration.
- For each and every push –pull type service, there will be a pre-defined message format along with a keyword, by which subscriber can request for content.
- The system will validate the incoming message against the pre-configured format. If the message format is not valid, considering all Services of the specific SC, a default response will be sent to the subscriber. The default response will be defined during SC configuration.
- For any relevant Key word of SC, the system will identify the particular Service based on combinations of SC and Keyword. SC-Keyword combinations for active Services only will be considered relevant.
- The system will then rout the request to the URL of relevant CP of that Service.
- Relevant CP will be configured during configuration of Service.
- A pre-defined default response will be sent to subscriber if system could not connect to relevant CP.
- If the CP is connected, the response will be validated and maximum 160 characters response will be sent to subscriber.
- There should be a common ID for MO and MT of a push pull request for reference.
- All keywords should be validated in Subscription Engine.

Register/de-register Mechanism

Figure 3: How Register/De-Register Happens
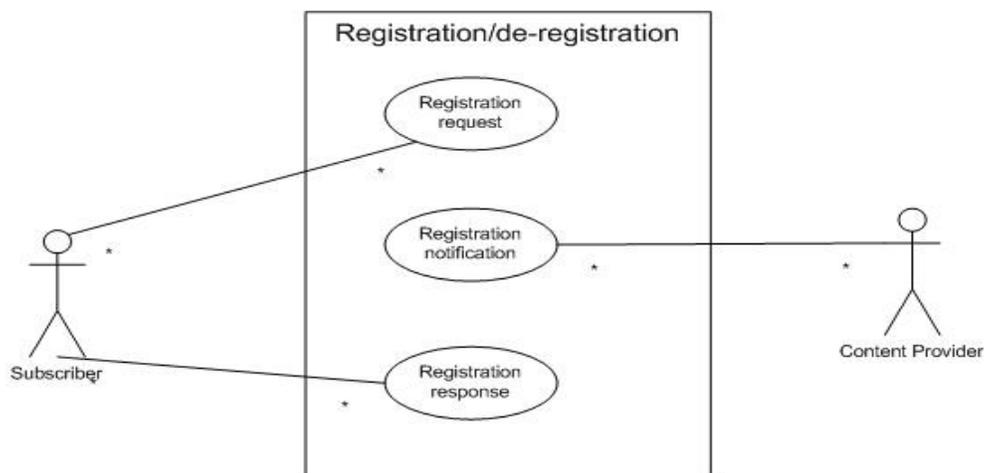
Subscription is a type of Service to which subscribers can register and they will be getting SMS contents in a regular interval. They also can de-register from the Service any time to stop getting contents. Therefore registration and de-registration is one part of this type of Service and getting actual contents in a regular interval is another part of the Service. This section will describe registration and de-registration part and the next section will describe getting actual contents of the Service, which we call Broadcasting of Subscription.

- The system will be able to identify whether the subscriber request for registration or push pull content requests (fetch from CP end) depending on pre-defined message format. This message format will be defined during Service configuration.

- For each and every broadcasting type service, there will be a pre-defined message format along with a keyword, by which subscriber can register or de-register to or from the service.
- For registration request, the system will add the requesting subscriber to a Service's subscription table against the particular port and keyword.
- For registration request, the system will check whether the subscriber is already subscribes for the particular service or not.
- A default response will be sent to subscriber for duplicate registration request and also sent to subscriber for successful or unsuccessful registration request. The default response will be pre-configured during Service configuration.
- For subscription cancel request, the system will set a flag for the requesting subscriber as a unregister user to subscription table against the particular port and keyword.
- For subscription cancel request, the system will check whether the subscriber already subscribe or not. If not system will sent a pre-defined default response.
- For subscription cancel request, the system will check whether the subscriber already unregister for a particular service. A pre-defined default response will be sent to subscriber for duplicate unregister request.
- For subscription cancel request, the system will sent pre-defined default response for successful and unsuccessful cancel request.

Broadcasting Mechanism

**Figure 4: How Broadcasting Happens**
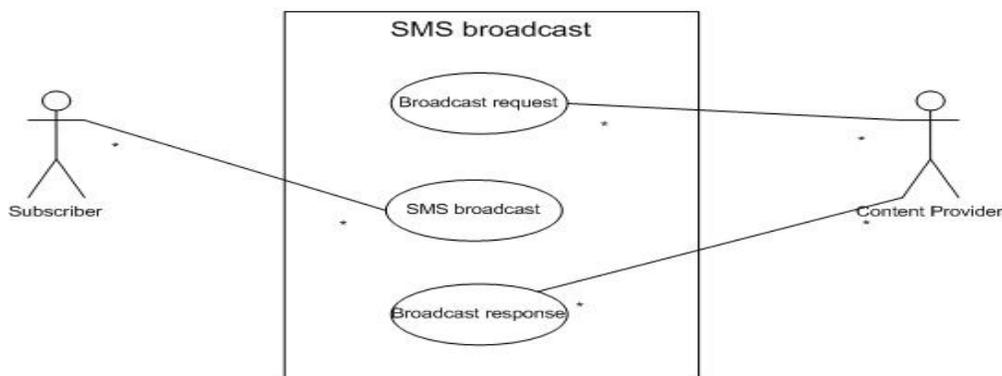
For Subscription Services, corresponding CP will send contents to the subscribers through the system in a regular basis.

- A CP can push Content against a Subscription Service which is already configured in the platform and the status of which is active.
- For this purpose an API will be exposed to the CP where CP will send the content with proper credential and format.
- After authentication and validation of the request, system will provide an acknowledgement to CP system mentioning size of sub-base of the Service.
- After authentication and validation of the request, system will broadcast the content to full sub-base of the Service.
- Supported content type would be: wap push, text, logo

5. Subscription Engine with improved Security system

As already discussed in the section 3 on the existing security system, now a improved architecture will be discussed.
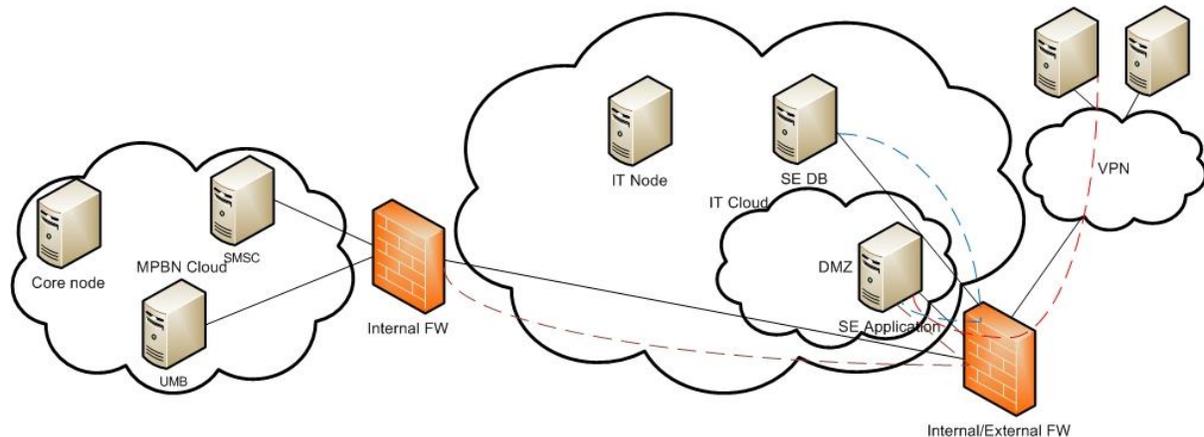
Here core systems and IT systems are hosted in different clouds. One is mobile packet backbone network cloud and another is IT cloud. Basically two different clouds are considered to ensure more security to the core part of the network. SMSC, UMB, HLR, MSS etc core nodes are hosted in the MPBN cloud. Enterprise systems like Business intelligence, CRM, Mediation systems etc are hosted in the IT cloud. Subscription Engine (SE) is an IT system that's why it hosted in the IT cloud. The application part of the subscription engine is hosted in the demilitarized zone that is also called DMZ. In this zone the API of subscription engine is exposed to internet. The content providers IPs are needed to be allowed in the External firewall to communicate with application of subscription engine. The application of the subscription engine will communicate to its database which is hosted in the internal IT cloud through internal firewall. That means two level of security has ensured in this architecture. Also the subscription engine's application needs to communicate with SMSC through two level of security checking. In this architecture if the application of the subscription engine is hacked then the other two firewalls will ensure security to the core nodes and other nodes in IT cloud.

6. Discussion and Conclusion

In this paper we have addressed the problem with existing traditional system and security systems and how we can improve this in an efficient way. Subscription engine is a customize version of service delivery platform (SDP) with a lower cost and improved performance. This is a solution to capture the content services of any operator through a single screen. In the lab we have tested and found it delivers high TPS approximately 700 to 800 TPS. So it shows better performance to cover a huge subscriber base within few minutes by delivering mobile terminating SMS. Also we saw significant improvement in case of customer complain. This reduces 30 to 40 percent in some cases.   In the security system which is mentioned here also ensures the end to end protection of the core, service and IT nodes.   This design is cost efficient, reliable and a secured solution. The significant of this study is focusing to deliver high performance solution with better customer perception.

References

Huawei, 2014, 'Service Delivery Platform', viewed 2 December 2014, http://www.huawei.com/en/solutions/arpu-up/hw-001252.htm

Metaswitch Networks, 2011, 'Service Delivery Platforms: New services made easy', viewed 2 December 2014, <http://www.metaswitch.com/sites/default/files/Metaswitch-White-Paper-Service-Delivery-Platforms-R1.pdf >

Net Cracker, 2014, 'Service Delivery Platform', viewed 17 November 2014, <http://www.netcracker.com/en/products/it-platforms/service-delivery-platform.html>

OMA, "The Open Mobile Alliance", viewed 6 December 2014,
 <http://www. openmobilcalliance.org/ >

The SDP Alliance, 2006, 'A Service Delivery Platform for End-to-End Value Web Enablement' 6 December 2014,
<http://www.thesdpalliance.com/downloads/SDP%20for%20End%20to%20End%20Value%20Web%20Enablement.pdf>

Teligent, 2014, 'Next generation service delivery platform', viewed 17 November 2014, <http://www.teligent.se/service-delivery-platform.html>

Wikipedia, (2014) 'Service Delivery Platform', viewed 7 December 2014, <http://en.wikipedia.org/wiki/Service_delivery_platform>