Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 8, August 2014, pg.669 – 678

RESEARCH ARTICLE

A Lightweight Access for Hybrid Mobile Web Cloud Content Architecture

Shawkat K. Guirguis, Adel A. El-Zoghabi, Mohamed A. Hassan
Alexandria University, Egypt
Institute of Graduate Studies & Research
Department of Information Technology
August 5, 2014

Abstract
The web was first design to just provide information that could be hosted over traditional client server model, the fast growth of web content and large numbers of web content today, trend to utilize the cloud computing and hyper mobile web, which provide instant computing power, scalability, availability, saving time and administration effort. Today there is an increasing demand for accessing the Internet from mobile devices, which becoming very popular, mobile web access now is an integral part of our lives, as the majority of current web content ignore the mobility, which considered as a great challenge for web content creators.

The main requirement when talking about the context of web future, is to enhance both features of cloud computing and mobile web content, by achieving the hybrid mobile web cloud content this allow us faster access technique, and enjoy with most benefits for mobile mashup for cloud computing. The main contribution of this paper is to combine the mobile web with cloud computing, to introduce an innovative computing model, called mobile cloud computing. By implementation the proposed architecture style experimental results show that the access response and excitation time is decreased, we gain a minimized transfer data size, and strongly utilizing the three screen vision view.

Keywords
Cloud Computing, Mashup Hybrid Web, Hybrid Mobile Web Applications, Mobile Applications, Mobile and Cloud Computing, Multi Tenant Solutions, Mobile Mashup, Enterprise Cloud Mashup.

1. Introduction

Today accessing web from mobile devices is becoming increasingly popular, now using mobile devices is becoming an important part of our daily life [1], in the early days of the web there was neither cloud computing technology nor mobile web access [2], also the concept of hybrid web applications was not introduced yet [3], in addition the numbers of published dynamic web content increase very fast, there is benefits for running cloud computing and hybrid web applications, as mobile technology and cloud computing running very fast the researchers try to find a solution to make using hybrid mobile web application or cloud mobile web content more powerful, enabling quickly build, deploy and offer better management.

The combination of wireless communication technology, cloud computing, portable computing devices and the concept of mashup has laid the foundation for a novel computing model, called mobile mashup cloud computing or hybrid mobile web cloud applications, which allows users an online access to unlimited computing power and extra powerful web applications [3], this type of combination can unveil a world of new innovative mashups technology that tackle some old problems, like challenges include how to abstract the complex heterogeneous underlying technology, how to achieve optimal adaptation under different constraints, how to model all the different

parameters that influence the performance and interactivity of the application, how to integrate computation and storage with the cloud while preserving privacy and security.

Architecture style introduced a general purpose model for hybrid mobile web cloud content, which could be more popular and easy to be used by most webmasters [4], web designers and offer a fancy mobile mashup for cloud computing, as the most web contents model is moving very fast towards Content Management Systems (CMS) web applications. Previously we introduced the lightweight architecture for mobile web content access (LAMWEC), this was a part of large work and now the proposed architecture style is considered as an innovative contribution that will help moving the world into faster mobile mashup web content implementation and publishing.

LAMWEC will provide us with more stable web with extra uptime, remove the administrative burden, and simplify mobile web site content development through employing the hybrid web application technology using cloud model that will host the application core, and weaving this with the traditional servers that will host separate unique web contents, to offer a fancy mobile mashup cloud computing or hybrid mobile web cloud content with faster access technique, the main idea is to have an architecture style that is not too complicated for casual users, and powerful enough for professional users. LAMWEC try to address a proper architecture style to provide integration between all the previous issues and to offer the solution in one single CMS system.

CMS kernel core system will enjoy with the huge power of cloud networks, architecture style allow you to enjoy the cloud power for system core for server side, and access web content from normal desktop or mobile device, with best treatment for mobile devices. We aim to set new custom rules for enhancing mobile web content accessing that run from cloud computing system [5], this is to optimize the data transfer size to provide a new mobile content web browsing model, good response time, offering scalability option and add instant power model.

LAMWEC Architecture style propose a dynamic model for web content delivery which divided into a kernel and GUI layer, that distribute resource over a cloud computing system [6], special enhancement of template engine system through using the natural of http application protocol, in this paper we will discusses the architecture style and its subsystems then illustrates the subsystems design model, the dynamic model will provide mobile web user with a faster mobile mashup cloud computing model, that based on build the GUI content over cloud computing system from shared kernel parameters, modify the CMS based template engine that will be based on invoking special kernel parameters for http response.

As the kernel CMS system core will be hosted under main cloud server, so the implementation of multi tenant software solutions will be available to easily handled here, the system administrator and web master could be select which is better for him, from multi tenant with single DB or multi tenant with single multiple DB this is one of power points of system flexibility, hence we try to address the problem of having multiple copies of the same application.

LAMWEC implement a new enhancement data movements technique in a single transfer, distribute media resources over CDN cloud system to minimize the bandwidth and access time [7], to make web pages access faster and increases the uptime access, the architecture style is designed to provide a majestic way, for facilitating web content industry using mashup for cloud computing system, and introduce the concept of mobile first.

We focus on how can we have a new architecture style for mobile cloud web content [8], that address the problem of mobile mashup and solve some problems like servers up time, optimize system administrator's effort, minimizing duplicated tasks, enhancing data transfer size for wireless channel, and provide new mobile web content browsing model.

The paper contribution is that to introduce an architecture style, for mobile mashup web cloud content that will enable your mobile web application to be developed, deploy, managed, administer or published easier and faster than previous, now we are able to deliver infinite scale so we can meet the increasing of hosted new mobile web content using the same multi tenant solution, a special media and images treatment that offer lowest possible latency for mobile users will be provided here, architecture style provide the mobile web content with a new enhanced features, and enjoy with a new benefits using hybrid mobile web cloud content, hence there must be an architecture to deliver an optimized mobile web content, and provide mobile users special treatment for web content access.

The rest of the paper is organized as follows, in section 2 literature survey, section 3 we discuss background topics for exploration, 4 we discuss the proposed architecture style and how it could help the web publish industry, section 5 architecture style model definition, section 6 conducted experiments, finally section 7 is the conclusion and future work.

2. Literature Survey

There were many researches that offer contribution for a mobile web or hybrid mobile web cloud applications, but most of them provide solution for specific points, there were a few contributions that provide a whole solution or complete one for hybrid mobile web cloud applications, the combination for all of the above technologies could laid the foundation for a novel computing model which could be

called hybrid mobile web cloud applications, research has not given the required attention to the development of mobile web content over cloud computing model.

Here we will discuss some approaches and their drawbacks. The first approach is an elastic web server farm for cloud based on Hadoop [9], using cloud serving infrastructures to host a web application to providing application level scalability, demonstrates its ability to run in isolation different web applications and scale dynamically on a cluster of machines, to run web applications that acquire and pre-process high frequency web feeds such as breaking news and finance quotes, the drawback was it solve the complicated nature and operational overhead of bootstrapping only, scalability for processing and storage only, but the application size remain as it.

The content delivery network system based on cloud storage [10], cloud storage service has recently emerged to provide content storage and delivery capabilities. Cloud storage adopting distributed storage technology and the cache technology, the system provided by telecom operators has mirror servers across the country and can automatically determine the most suitable route between the mirror server and the WAP service users, this model provide storage only, and no integration between application server and storage server. A cloud service environment framework for SCORM compatible content [11], it offer deeply influenced e-learning area, it's a proposal for cloud service environment framework and its service mode compatible content, it realizes content's unified storage and the freely sharing of learning content and record for outside applications, which breaks traditional SCORM compatible LMS's limitations and reduces storage and management cost of SCORM compatible content.

And for hybrid web cloud applications there were some approaches were interested in web mashup and its tools [8], mashup composition tools are at the core of emerging software engineering paradigm, benchmarking is a promising approach providing a strong evaluation mechanism based on quantitative and reproducible measurements, the goal here is to delimit the scope and discuss the feasibility of a unified benchmarking framework targeted for Web-based mashup tools. An efficient mashup tool for searching and fusing web entities [12] this demonstrate a new mashup tool to search and integrate web data from diverse sources and domain specific, it supports adaptive query sets of relevant entities with a communication overhead, it implementation supports a high degree of parallel processing, in particular a streaming of entities between all data transformation operations facilitating a fast presentation of intermediate results.

Last approach is for mobile mashup application [13], it describes a mobile web application that allows browsing conference publications, it queries a main endpoint that serves the conference metadata set, it can be linked to external web services, it follows recent W3C technical advances and as a mashup, requires few server resources, but it was much tied to specific application type and not large enough to be considered as a framework or architecture. Our proposed new architecture model will try to avoid the problems that face the previous approaches, and try to take advantage of the others [14], to provide a very unique contribution for hybrid mobile web cloud applications that and provide a lightweight for mobile web content access over enterprise cloud model.

3. Topics for Background

3.1. The two Extremes of Mobile Applications

The foundation of mobile cloud computing, comes from the combination of cloud computing, portable computing devices, wireless communication, location based services, mobile Web, etc. [15], the mobile devices is considered as the entry points and interface of cloud online services, and allow users to access to unlimited computing power and storage space within the mobile domain, there are many hurdles to overcome in mobile cloud computing.

Mobile devices inherently have and will continue to have limited resources as processing power, memory capacity, display size, and input forms. These have been the forming factors of existing mobile application approaches [16]. This is why mobile computing is characterized by severe resources constraints and frequent changes in operating conditions, to identify the fundamental challenges in mobile computing there are several researchers in this area, there are two main forming factors of existing mobile application approaches, we believe that the full potential of mobile cloud applications lies in between these two extremes:

3.1.1. Mobile clouds native or native offline applications:

Most of modern devices applications fall into native offline applications, periodical synchronization between client and backend system is the main issue here, this model act as fat client because the processing of the presentation and business logic layer, is locally done on mobile devices with data downloaded from backend systems, most resources available locally on the fat client, rather than distributed over a network as is the case with a thin client.

3.1.2. Web applications or online applications:

This application model requires that the connection between mobile devices and backend systems is available most of the time, this application type is based on the web technology that has a powerful alternative to native applications, the problems of a cross platform issues the web technologies can overcome them, the advantage of this model overcome some of the disadvantages of offline applications, but it has its disadvantage also.

3.2. Mashup is the Way of the Future

When the core functionality of web or mobile application requires mashing up services, this is considered as a mashup application [17], mashup is creating a new web page or application from multiple sources, based on combining resources or functionalities from two or more sources, to run a new application or service, there are many examples of mashup which used for many different business types, for example mashup video is a video that was edited and build from more than one source and finally displayed as a single one video, the most famous example which is very tied to our work is the mashup web application hybrid, which is the web application that its data and/or functionality is combined from more than one source, mashup has characteristics or number of components, that could be describe a sequence architecture model because all of them is based on each other in a sequence scenario, below is a list of this mashup components:

Type: Like Data, Consumer and Business mashup.

Service Provider: Which they provide access to their resources or services online, this provided services or resource construct the mashup web applications.

Protocol: A particular set of rules and guidelines to exchange messages, between services which offered by service providers, protocol used for communication between server and client, and between the resources and the server.

Data format: Describes how the information is stored in the exchanged messages, these messages contain the required information in a pre defined data format.

Architecture: How the transmission done to the client from the combined resources in the server, there are main different types of these architectures, such as directly send the information to the client or combining it in the server first before sending.

Interface: Last step in this sequence scenario is the client have to see final results from a web interface, all of this is based on the mashup type, which done in the first step.

3.3. Data Exchange to Provide Lightweight Format

The process of taking data structured under a source schema and transforming it into another data structured under a target schema this is called data exchange, by the end of this process the targeted data is an accurate representation of the source data to have citation needed, data exchange is based on restructured data unlike the related concept of data integration, there are many languages that could be use for data exchange, and this is called data exchange language and it can be used for any kind of data, this based on the capabilities and qualities determined by comparison with the capabilities of natural languages, any file format that can be read by more than one program could be classified also as a data exchange language.

Actually there are certain types of languages are better suited for this task than others, because of their specification and natural, there are many popular data formats, some of them like XML and JSON is the best is some applications like mashup, because they are designed to enable the creation of dialects, also they are the most practical choice on internet and web applications today. Mashups could be classified to Rich Internet Applications (RIAs) classification [8]. This class is famous with the lightweight because it use the XML and JSON for message exchange, and this is one of its very important characteristics, this fact led mashup to be quickly development, good enough and situational application developments, it allow applications to have a rapidly changing needs responding, traditional web applications development rely on careful and relatively long lasting development processes that is the opposed to mashup development.

3.4. Cloud Computing and Grid Computing

Cloud Computing is the latest paradigm that involves delivering hosted services over the Internet. Cloud is the most hyped word in the world, and everyone in the industry has his own definition. Cloud computing lets you access all your applications and documents from anywhere in the world, freeing you from the confines of the desktop and making it easier for group members in different locations to collaborate. Cloud computing is a new promising technology that is about how we store information and run applications, instead of running programs and data on an individual desktop computer, in cloud everything is hosted in the cloud network, which is collection of computers and servers accessed via the Internet, we expect the universal access, 24/7 reliability, and ubiquitous collaboration promised by cloud computing [19].

Cloud computing often confused with grid computing, but there is difference between them, when you apply the resources of numerous computers in a network to work on a single problem at the same time, now you are using the grid computing, usually this implementation is used to address a scientific or technical problem that often needs more processing power [20], the famous example for this is the Search for Extraterrestrial Intelligence (SETI), people all over the world allow the SETI project to share the unused cycles of their computers to search for signs of intelligence in thousands of hours of recorded radio data. Berkeley Open Infrastructure for Network Computing (BOINC) is another example of grid computing example [21]. In grid computing it's necessary to use software that can divide and then send out pieces of the program to thousands of computers, this can be done throughout the computers of an organization, or it can be done as a form of public collaboration [20].

3.4.1. Cloud Service model

In the next decade cloud computing we will promises new ways to collaborate everywhere, through mobile devices, most business applications are moving to the cloud. It's not just a fad the shift from traditional software models to the Internet has steadily gained momentum over the last 10 years [22], there are many different cloud services guises models that would serve you [23], figure 1 illustrate cloud service relationships for IaaS, PaaS and SaaS.

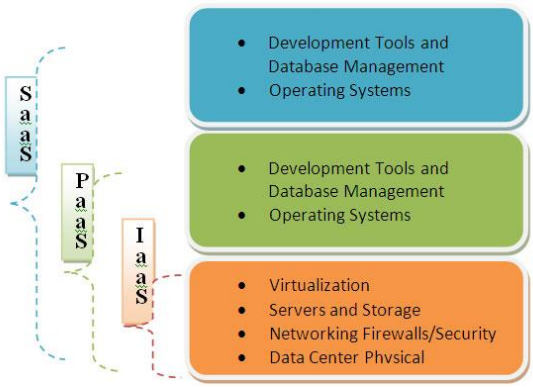


Figure 1: Cloud Service relationships for IaaS, PaaS and SaaS

Cloud services are the basic principle behind the emergence of cloud computing, figure 2 illustrate cloud service and cloud deployment models, the main categories of cloud services that can be offered by any cloud architecture and technology is [24]:

- Infrastructure as a service (IaaS)
- Platform as a service (PaaS)
- Software as a service (SaaS)
- Network as a service (NaaS)
- Hardware as a service (HaaS)
- Data as a services (DaaS)

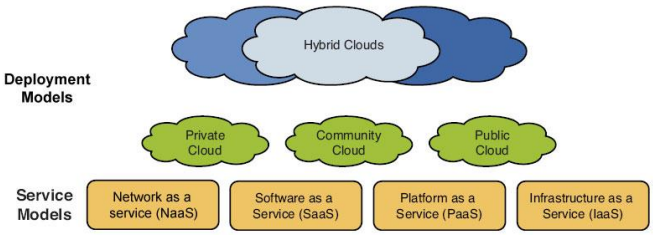


Figure 2: Cloud service and cloud deployment models

4. Proposed Architecture Style

Hybrid mobile web cloud content or mobile mashup cloud computing is the core for architecture style, as hybrid web application is based on mix web content from multiple source, here we will divide the CMS into, application core section which will be called as a system kernel, and user defined section, the application core will be hosted under the main cloud server, and user defined will be hosted under the user traditional client/server, this is to shift the computation between cloud and traditional servers [25], there will be new type of caching also to add extra faster resource access, it's very clear that the closer you are the faster is.

The proposed architecture style is based on two main directions, the first is hybrid cloud web content, and the second is mobile web content enhancement, this two directions is the core of architecture style implementation, figure 3 illustrate the proposed architecture style design.

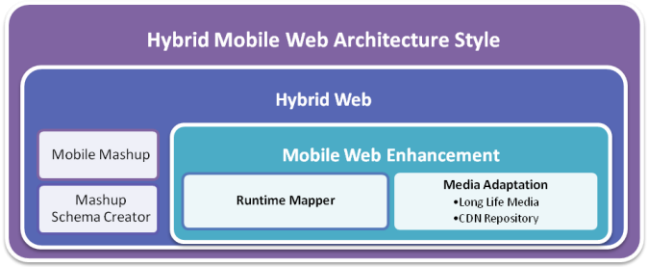


Figure 3: Proposed Architecture Style Design

Mobile adaptation is the key to mobility and we will use this in mobile web content enhancement [14], mobile web content enhancement style based on media size enhancement and data size enhancement, this is done by let web master create and write his web site content as he like, then there will be a convertor engine that will update the media content link to use the best media version for contacted mobile device, from nearest CDN server or using a long life images technique this is based on the type of cached that is configured here, and for data size optimizer the convertor and translator engine will start to convert the code that should be passed to run under client device, this to a new mobile code that will be based on using JSON to create a new browser content, this is to simplify the code for mobile users.

5. Architecture Style Model Definition

Our architecture style is based on CMS technology that is running over cloud computing model as a case study, and because that mobile devices today can be seen as entry points and interface of cloud online services, architecture style offer a new mobile browsing experiment using enterprise cloud mashup model, we try to minimized webmasters and administrators effort and try to solve most of their common problems, we try to move the processing computation to cloud model, and relax web admin from all common overhead, also we try to provide a new model that minimize the processing overhead and reduce the amount of transferred data over a wireless channel, the architecture style will provide also a new custom hybrid mobile web cloud content.

From point of view for multi tenant software solutions, architecture style could be implemented as multi tenant with single DB or multi tenant with single multiple DB, this is based on how site admin intend to develop his model, and this considered as power point of system flexibility, our

work goes through many directions, figure 3 illustrate architecture style roadmap protocol definitions, next we will explain briefly in a few lines about the role of each direction, we consider the following as the five main directions.

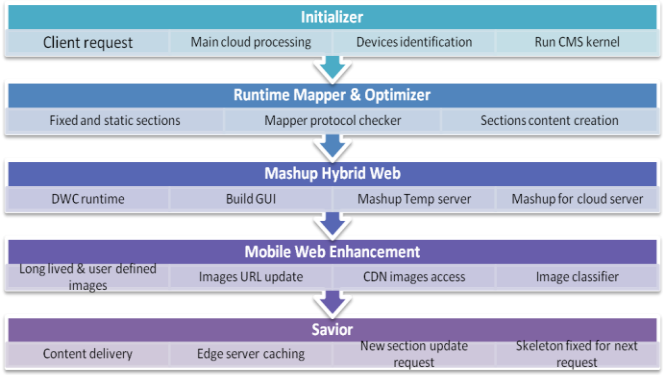


Figure 4: Architecture Style Roadmap Protocol Definitions

5.1. Mobile Mashup Initializer (Role of the road sign)

We accept client request the main cloud server process this request and try to identify the connected device, which will help us to know the connected device type, what its screen size, we will have a special treatment for the three screen vision concept to create custom web content for current connected device, current web content that is related to this request should be passed as a kernel parameter to the shared running kernel, this should run some other kernel parameters to get some other information, is this the first or second request and some other kernel parameter that will help in the next phase, and also what is the current running model for multi tenant.

5.2. Runtime Mapper & Optimizer (Role of the optimizer)

Shared kernel is ready to generating the web content and build GUI, architecture style will allow template creator to define and partition the template into small parts, each part must belong to one of the following classification sets, Rarely or Sometimes or Always, the result of this all is that we will have a mixed running model that will provide us with the final GUI layout, from a fresh dynamic running parts and complied parts, this will help in optimize running process and then save processing time, this enable running a full dynamic web pages with a better performance like if it was a static pages, this was impossible without this smart solution, kernel creates the response with new data formats for data exchange that will provide lightweight format.

5.3. Mashup Creator for Hybrid Web (Role of the mentor)

One of the power points of this contribution comes from that architecture style enjoy with most cloud computing benefits, and add special treatment for mobile mashup hybrid web model, now based on requested DWEC and current connected device mashup main component ready to mashup current web content, mashup creator is responsible for map current request to current template for running DWEC, and manage the presentation layer with its corresponding database, most kernel parameters and roles will done here in order to make final response ready to send back to the client, here is an important part which how the current content will be ready for next phase which is mobile web enhancement.

5.4. Mobile Web Enhancement (Role of the postman)

mobile web enhancement include many other subsystems all of them is responsible for optimizing and enhancing data size for transferring them over the wireless channel, the implementation for long lived and user defined images concept comes here, the images classifications will be skeleton images resources and user defined resources, user defined will be classified into sub class which is semi persistent and volatile persistent, shared kernel now should update the images resources URL and deiced which type of image classifications should be used, cloud help running shared kernel to check where is the nearest CDN repository that should be used, this will enable faster response to client as the client now will request the image resources from the nearest edge server from him, best version of image should be passed to the client, based on current connected device and what is the best version for this device, shared kernel now will update all above mentioned parameters and set final image URL that should be used with current web content.

5.5. Master Savior Generator (Role of the savior)

Based on the type of implemented multi tenant software solutions for current shared CMS version, master savior generator which will combine all previous subsystems and integrate them to create final response, as generating web content for mobile is not simple like the desktop most of mobile mashup should be integrated here, now we are ready to send the final web content back to the client, caching under nearest edge server should be done now for facilities next client requests, also next request will have to update new parts only and keep any cached parts that has no new update, so new requests will keep the main skeleton as it if there is no new update happen on template server.

The architecture style cloud section has two main components, server component and cloud component, the CMS kernel will be hosted in the main cloud server, each web content presentation layer (PL) in addition to its database, will be hosted under the traditional web server, each PL with its database produce a version for web content for current web content, each GUI has a template sets that produce different content design interface for some current used design.

This defining formal notation for dynamic web content that uses same kernel over some cloud servers:

$$\exists\ cp,g\ s.t.\ \forall\ w \in W \mid W = \{w_1, w_2, w_3, \dots, w_n\}$$

Where D is all dynamic web content that hosted over traditional web server and use the same kernel from this main cloud server, cp is the current presentation layer with

for some current of database version, and g is the GUI with a its current template.

This is defining formal notation for different web content sets, with its PL and GUI template set:

$X = \{x_1 , x_2 , x_3, \dots , x_n\}$, $P \{p_1 , p_2 , p_3, \dots , p_n\}$, and $T \{t_1 , t_2 , t_3, \dots , t_n\} \mid P \subseteq A \,\&\, T \subseteq A$

X is a set that contain different dynamic web content (DWEC), which is running under different traditional web servers, that uses the same CMS kernel over the main cloud server, P is a set of presentation layer interface and its database content, T is a set of template set, such that P is a proper subset of A and T is also a proper subset of A.

6. Experiments

We conduct different six experiments, that will provide to us the evaluation of the architecture style and proof that we have complete the architecture style successfully and give us a conclusion about our work, each scenario of this experiment provide a new measure.

Mobile	OS	Display	Processor Speed	Internal Memory	Display Color
iPhone 5s	iOS 7	4.0 inches	Dual-core 1.3 GHz	1GB	16M
LG Nexus 5	Android OS, v4.4	4.95 inches	Quad-core 2.3 GHz	2GB	16M
Samsung Galaxy S4	Android OS, v4.2.2	5.0 inches	Quad-core 1.6 GHz	2GB	16M
Nokia Lumia	MS Phone 8	4.5 inches	Dual-core 1.5 GHz	2GB	16M

Table 1: Different emulators with different parameters that used for testing

We use different mobile emulators trying to cover most common mobiles devices used today, table 1 explain some devices used in this experiments, our contribution here is to explain how can we offer hybrid mobile web cloud content, that will work well in most mobile device in different parameters and hosted over cloud computing model.

6.1. Mashup for Different GUI use the Same Kernel

This experiment will going to evaluate how we could offer mobile mashup cloud computing, for different GUIs with different web content that use the same shared kernel, all of this created instance will enjoy both of advantages for cloud computing and mobile mashup web content, like allow for reusing of existing applications, time saving and rapid development, no extensive IT skills are required and cheap cost efficient.

We access three different web content each one has its own database, GUI interface and its custom configurations, all this web content instance is hosted on different dedicated servers, but all of them use the same cloud shared system kernel, this experiment is created using a normal computer, with a Google Chrome web browser running on a windows operating system, with a resolution 1440 * 900 and this is considered a large screen size in our testing, we only focus on the multiple instance building feature, figure 5 illustrate how could we display the home page of different three web content, each of this instance implement its own GUI, database content and its custom configuration.

Figure 5 Multiple instances web content running the same shared kernel

6.2. Content Access Response Time

In this experiment we access different three mobile cloud web contents from the same mobile devices, this explain how we get more benefits and features to mobile web content creation industry, and content publishing through optimizing the transferred data size and allow faster mobile web content access, we access both the same web content versions, the traditional web content which run over the traditional CMS, and same content layout after migrating it to the cloud mashup model, which run the same shared CMS system kernel, figure 6 illustrate the response time for same web content that run over traditional model compared with hybrid mobile web cloud content access response time.

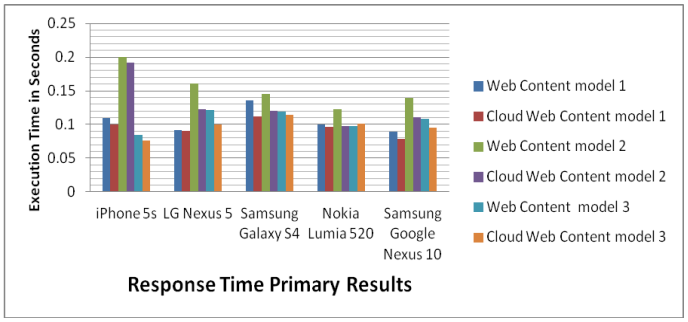
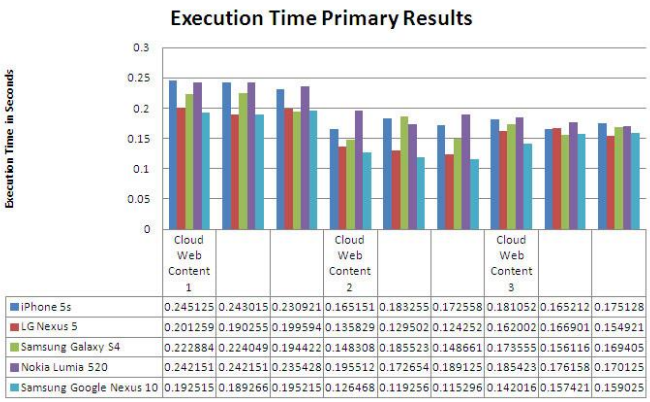


Figure 6 Primary Results for Response Time

6.3. Hybrid Mobile Web Execution Time

In this experiment we access different three web contents instance from different five mobile devices that are listed in table 1, we create different three experiment when access the same web content from the same mobile device, this offer us a three series sets for each mobile cloud web contents, figure 7 illustrate execution time primary results, we try also to check if there is extra overhead or another new load from our main subsystems, which they are running over CMS engine to implement architecture style.

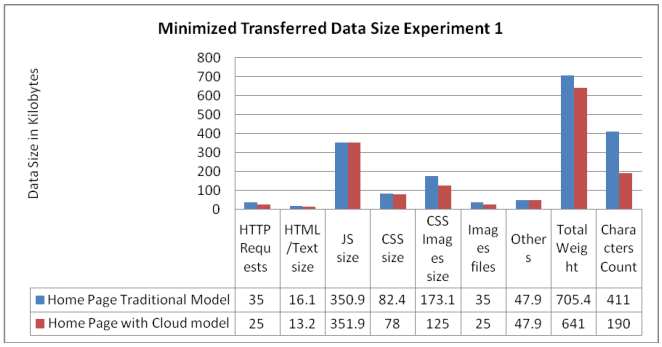


	Cloud Web Content 1			Cloud Web Content 2			Cloud Web Content 3		
iPhone 5s	0.245125	0.243015	0.230921	0.165151	0.183255	0.172558	0.181052	0.165212	0.175128
LG Nexus 5	0.201259	0.190255	0.199594	0.135829	0.129502	0.124252	0.162002	0.166901	0.154921
Samsung Galaxy S4	0.222884	0.224049	0.194422	0.148308	0.185523	0.148661	0.173555	0.156116	0.169405
Nokia Lumia 520	0.242151	0.242151	0.235428	0.195512	0.172654	0.189125	0.185423	0.176158	0.170125
Samsung Google Nexus 10	0.192515	0.189266	0.195215	0.126468	0.119256	0.115296	0.142016	0.157421	0.159025

Figure 7 Execution Time for Three Instances Using Same Shared Kernel

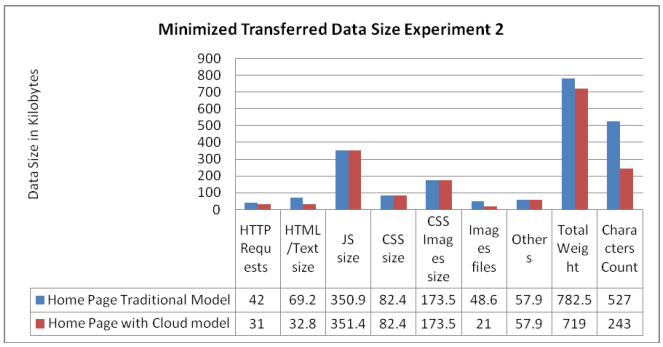
6.4. Minimized Transfer Data Size

We access both home page and inner page for single one implemented instances, from Samsung Google Nexus 10 this create to us two main experiments, this device is considered as an intermediate mobile device, which located between desktop and smart phones mobile device, also it's the intermediate device in our three vision screen, we have measure how the mobile web enhancement play a great role in optimizing the data size, different parameters like numbers of HTTP requests, the HTML text size, JS size, total transferred size and also some others options, we run this two experiments with traditional web content and with the same web content, after implementing the new architecture for different two pages, we will differentiate between first request and second request by the transferred data size, figure 8 and figure 9 illustrate the analysis of minimized transfer data size results for mobile mashup enhancement, with data exchange to provide lightweight format for desktop and for mobile devices, the transferred data size in kilobytes when access web content from mobile device.



	HTTP Requests	HTML /Text size	JS size	CSS size	CSS Images size	Images files	Others	Total Weight	Characters Count
Home Page Traditional Model	35	16.1	350.9	82.4	173.1	35	47.9	705.4	411
Home Page with Cloud model	25	13.2	351.9	78	125	25	47.9	641	190

Figure 8 Primary Results for Home Page to Analysis Minimized Data Size Transfer after Mobile Web Enhancement



	HTTP Requests	HTML /Text size	JS size	CSS size	CSS Images size	Images files	Others	Total Weight	Characters Count
Home Page Traditional Model	42	69.2	350.9	82.4	173.5	48.6	57.9	782.5	527
Home Page with Cloud model	31	32.8	351.4	82.4	173.5	21	57.9	719	243

Figure 9 Primary Results for Inner Page to Analysis Minimized Data Size Transfer after Mobile Web Enhancement

6.5. Media Adaptation for Long Life Media

In this experiment we request a three different web contents, to measure the media size used for each web content and show how media adaptation offer a great optimization value, each web content has two sub scenarios first before apply the architecture style, which have not any media adaptation treatment, and the second scenario after applying the architecture style which provide us with long life media concept, we compare both two sub scenarios with each other's and also check the difference between data sizes for each, we will go to measure the size of CSS images files and normal images files, all of this two types is classified to skeleton images, semi persistent images and volatile persistent images, in first scenario there is no effect for this three types we just list how much their size was, but in second scenario we show how we use befits of this classifications and how they effect in data transferred size optimization, and how this save media size for next request after applying the architecture style, figure 10 illustrate the media adaptation with long life media enhancement for mobile mashup cloud computing.

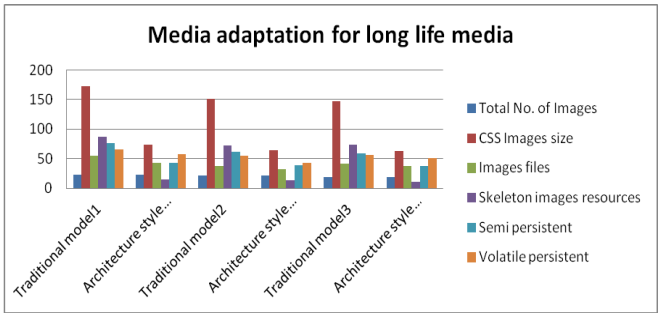


Figure 10 Media Adaptation for Mobile Mashup Cloud Computing

This give us a real experiment results which will proof that we can offer a new model for hybrid mobile web cloud content and having many different version for web content that is based on one single kernel that is hosted over the cloud computing, without having to create a new copy for each new web content, and host is separately on client server model, all of this without adding more layers to the system which is not perfect all the time.

6.6. Experiment for Subjective Test

The first experiment is evaluation for webmasters subjective test for using shared kernel for mashup building, this measures could provide validity to architecture style, and for evaluation of hybrid mobile web content, we create a two new test which is the subjective test.

We get webmasters opinions about using the architecture style for administer their web contents, and how this style add more value and features for using the new era of mashup CMS cloud model, we conduct a subjective test with 40 webmasters, the rating system scale was 0:10, 0 is very bad and 10 is perfect, we create two tests the first one is for webmasters administration working effort, and the second one for webmasters evaluation for architecture style adding features to have a full opinion about the overall proposed architecture style, figure 11 illustrate admin effort after use architecture style with its administration area, figure 12 illustrate webmasters features evaluation after using architecture style to implement and create a new content.

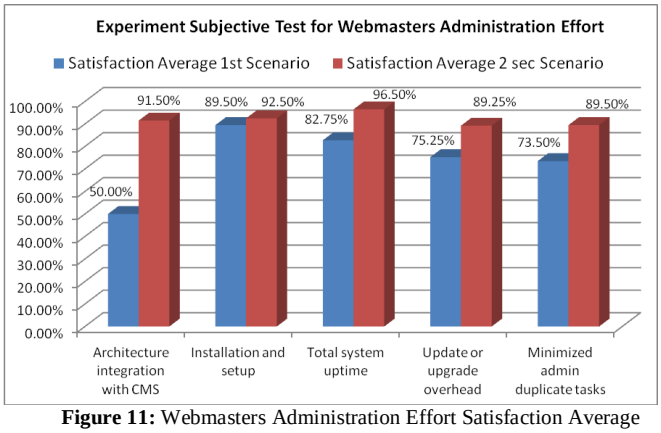


Figure 11: Webmasters Administration Effort Satisfaction Average

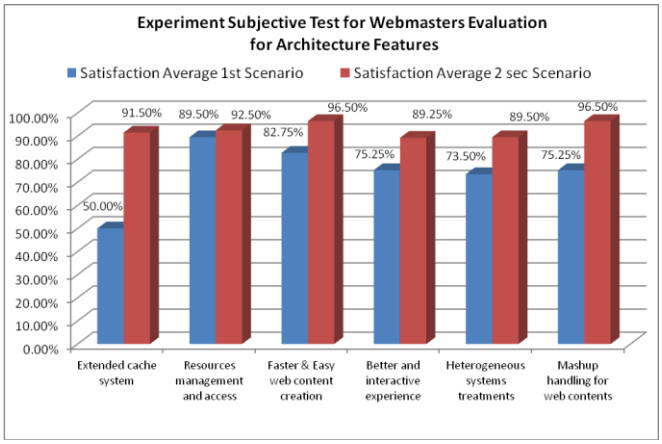


Figure 12 Webmasters Evaluation for Architecture Features

7. Summary and Conclusion

The aim of this work is to introduce a new innovative architecture for hybrid application to build mobile web content, in order to fulfill and to address the problem of having multiple copies of the same framework, which used to create multiple web content, and enhancing the mobile web content browsing, the architecture style solution provide an enterprise web platform architecture with a very optimized network data transfer, enabling scalability, instant computing power, save time and administration effort, offer a smart technique for optimized access over http protocol, separation between system kernels, logic and interface will enter our architecture style to the new era that best matching cloud computing technology.

During this paper, we studied the hybrid mobile web cloud application that uses the same shared kernel, how special CMS clustering application model being employed, this makes us trying to find an applicable solution for mobile web content access over enterprise cloud mashup, and we try to find a novel architecture style solution that provides enterprise mobile web platform architecture with an optimization enhancement for network data transfer. Our principal contribution is not only to implement all the previous ideas but put all of them in a single package with a very high coherent to each others. From this point of view we developed the lightweight architecture for mobile web content access over enterprise cloud mashup.

This proposal should move the world into faster web content implementation and publishing, and will enable webmasters and site administrators to quickly build, deploy and manage their web contents, also it will offer new web applications opportunities, the LAMWEC use most of the cloud computing benefits like, elasticity, economics, all services is automated, all services is managed and the most power point is that it's always up-always on. The LAMWEC focus on user experience and offer a real solution for it, better picture treatments for the three devices concept, offering a complete innovative solution for web content creation and publishing for hybrid mobile web cloud content.

References

[1] http://www.fool.com/investing/general/2013/01/19/mobile-overtakes-desktops-in-2013-microsoft-be-war.aspx, retrieved on 2014

[2] Web Services, Service-Oriented Architectures, and Cloud Computing, Douglas K. Barry, David Dick, 2013.

[3] Mashup: Web application hybrid, Stephen Baden, Drew Lefebvre, Jamal Dar, Mathew Langner, Min-Tse Yu, 2011.

[4] Deploying Customer Service in the Cloud: The four phases from implementation to transformation, by Oracle, 2012.

[5] "Mobile Web Development" © 2008 Packt Publishing, ISBN 978-1-847193-43-8

[6] A Study of Cloud Mobility in a Mobile Cloud Network based on Future Internet Approach, Dongha Kim, Hyunjun Kim, Gijeong Kim, Sungwon Lee, 2012.

[7] http://whatis.techtarget.com/definition/ content-delivery-network-CDN, retrieved on 2014.

[8] Towards A Unified Benchmarking Framework for Web-based Mashup Tools, Saeed Aghaee, Cesare Pautasso, 2012.

[9] An Elastic Web Server Farm for Cloud Based on Hadoop, Krishnan S., Counio J.C., 2010.

[10] The Content Delivery Network System Based on Cloud Storage, Yuedui Wang; Xiangming Wen, Yong Sun, Zhenmin Zhao, Tianpu Yang, 2011.
[11] A Cloud Service Environment Framework for SCORM Compatible Content, Wen Huang, Yi-Xia Zhao, Yun Xiao, Xing-Hua Sun, 2011.
[12] An Efficient Mashup Tool for Searching and Fusing Web Entities, Stefan Endrullis, Andreas Thor, Erhard Rahm, 2012.
[13] DataConf: Enriching Conference Publications with a Mobile Mashup Application, Lionel Médini, Florian Bâcle, Nguyen Hoang Duy Tan, 2013.
[14] A Smart Framework for Web Content and Resources Adaptation in Mobile Devices, Shawkat K. Guirguis, Mohamed A. Hassan, 2009.
[15] Mobile Cloud Computing: A Comparison of Application Models, Dejan Kovachev, Yiwei Cao and Ralf Klamma, 2011.
[16] Mobile Web on the Desktop: Simpler Web Browsing, Jeffery Hoehl, Clayton Lewis, 2011.
[17] http://en.wikipedia.org/wiki/Mashup_%28web_application_hybrid%29#Types_of_mashup, retrieved on 2014.
[18] Mashup: Web application hybrid, Stephen Baden, Drew Lefebvre, Jamal Dar, Mathew Langner, Min-Tse Yu, 2011.
[19] Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online, Michael Mille, 2009.
[20] Cloud Computing Vs. Grid Computing, Seyyed Mohsen Hashemi, Amid Khatibi Bardsiri, 2012.
[21] http://www.ibm.com/developerworks/library/wa-cloudgrid, retrieved on 2014.
[22] http://searchcloudprovider.techtarget.com/definition/cloud-services, retrieved on 2014.
[23] http://www.ibm.com/developerworks/cloud/library/cl-cloudservicemodels/?cmp=dw&cpb=dwcld&ct=dwnew&cr=dwnen&ccy=zz&csr=021011, retrieved on 2014.
[24] http://www.oracle.com/us/solutions/cloud/infrastructure/overview/index.html, retrieved on 2014.
[25] Lightweight Architecture for Mobile Web Content Access over Enterprise Cloud Mashup, Shawkat K. Guirguis, Adel A. El-Zoghabi, Mohamed A. Hassan, 2013.

