Facebook Distributed System Case Study For Distributed System Inside Facebook Datacenters

Asma Mohammad Salem

Department of Computer and Networking engineering, JU Amman, Jordan
asma_salem85@yahoo.com

Abstract: Facebook is recognized as the largest online social network system in the last few years, which is come up with billions numbers of users in the last 2013. The system is recognized as distributed system in its design, infrastructure and architecture .The datacenters behind this network system are huge, robust, keeping the system scalable, reliable, secure, and let the Facebook accessible from anywhere with highly availability

Keywords: facebook; distributed system; availabilty; scalability; Hadoop ; social cloud ;Hive ;HDFS ;

1. INTRODUCTION

Facebook system was founded in 2004, with a mission to give people the power to share and make the world more open and still connecting them with friendship relationship. People from anywhere can use Facebook to stay connected with friends and family, they can share such contents of data and multimedia such as audios/videos and express what matters to them by comments and likes [9][10]. Facebook systems at the first are responsible for processing large quantities of data, named as “Big Data”, which is ranging from simple reporting and business intelligence to the huge measurements and reports executed from different perspectives [8], this numerous large of data located on different geographically distributed datacenters and being processed under highly equipment’s servers, which they architected in high technologies to improve the whole performance of the system ,Facebook inspired by Hadoop and Hive systems [1][2] supported by its integrated components which Facebook was built on top of this technologies [3] [4] . We will go through the system details, starting with the system feature in section (2), exploring the system design and discussing all system components in details in section (3), as the system is being under steps of enhancements we will explore some of these enhancements in this section. Nowadays Cloud computing is the main topic for supporting systems and realizing applications. Facebook system is as a geographically distributed system is recently being integrated with its feature and services by cloud computing solutions. Ending the system design with cloud technology solutions, this paradigm shift in technologies would server an alternative solution that could keep system in Facebook dynamically scale in the future, and maintain the rapid growth while keeping performance metrics in bounds and saving the system stability and functionality [5][6][7]. Ending up with our conclusions in section (4), we will investigate the whole system design and implying our critical evaluation for this distributed system features and ,design.

2. DISTRIBUTED SYSTEM FEATURES

Starting at early system features and services, Facebook as an example of commercial Online Social Network (OSN), and a hosted application that attract users with a set of features and attracts advertisers, who pay for the privilege of displaying ads targeted to these users. OSN interconnecting users though friendship relations, and allow for synchronous and asynchronous communications of user generated content ,such as text ,multimedia from audio ,video and such third party OSN application updates over a social graph ,these services attract users and will be the main reason for huge traffic that flow through the system parts Facebook system is an open website that is published on the internet as a social network system, in which a user can easily connect to by accessing its home page and continue registration by few screens that navigate him or her within few steps to complete the registration. Accessibility to this system is provided by any device that has accessibility to the internet, these machines such as desktops, mobiles… etc. some of them listed in figure (1). Facebook features are being provided by starting with a registration phase, requiring a user name and password, this registration in done once and a login should be done after registration to start using the website, starting with inviting your friends from your email account, this website is based on 2 building a friendship to start sharing your status, media and news with them, most of website are:-

Wall: it is the original profile space for a user where contents posted there, including photos and videos, and files, user can attach any content on his or her wall and being visible to anyone, by choosing the space of visibility on the wall user can limit visibility to the wall contents, which were in early versions of Facebook as text only [9].

News Feed: it is a home page in which users can see a continue updates list of their friend’s activities. They can explore information that includes profile changes, updates and coming events, users can explore the conversations that taking place between the walls of a user's friends.

Timeline: a space in which all photos, videos, posts, and contents are categorized according to instant of time in which they were uploaded or created.

Friendship: this feature is what Facebook is based on, “Friending" someone is the act of sending another user a friend request on Facebook or accepting friendship request. A user has full control to manage his or her friend list.

Likes and tags: it is positive feedback; users can apply likes on updates, comments, photos, status and links posted by their friends, these likes make the content appear in their friend’s pages notifications and updates.

Notifications: keeping track of all the most recent actions or updates. It is an indicator to inform the user that an action has been added to profile page, his or her wall or time line, any Comment or like, shared media that being tagged in [9].

Networks, groups, and pages: Facebook allow users to build their networks ,groups and creating pages which combine them around an idea or specific community .they can used for posting items or issuing messages for a group of users who join these communities.

Messaging and outbox: a service allows users to send messages to each other. Users can send a message to any number of friends at a time. Managing messages also provided .By the year of 2010, Facebook announced a new Facebook Messages service which give a user an account under the Facebook.com ,This system is available to all of the users, providing text messages, instant messaging, emails and regular messages [2][10] .

All these features and more are being served on Facebook , adding the different applications that are : events ,market places ,notes ,places, questions ,photos ,videos ,and Facebook pages ,we are interested on the system features that will produce the traffic basically ,we will categorized them in few later lines in major categories in which they will help us in system investigations .this categories are based on the type of data and the communication mechanisms [9]

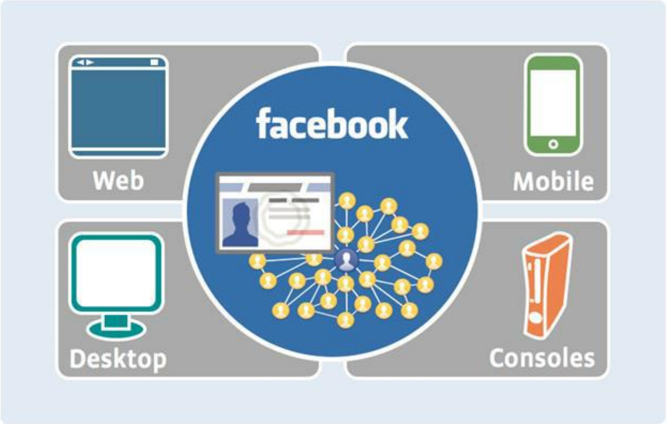


Figure (1): Facebook accessibility feature

OSN traffic patterns consist of Facebook built in interactions:

1- the wall post which is known by status updates: allowing users to share text and multimedia consisting of audio /video which is will published on the main page of their own pages or friend's profiles interface, and could be easily seen by a user friends and to be under their comments and likes of course, these updates are could easily come to the surface by pushing or polling the Facebook updates [9].
2- The comments and the like tags: Comments and likes are the second mode for interaction and being used on existing post and updates [9].
3- Facebook Messages and chat: a service allows users to send messages to each other. By the year of 2010, Facebook announced a new Facebook Messages service which give a user an account under the Facebook.com ,This system is available to all of the users, providing text messages, instant messaging, emails and regular messages, every user has a strong controls over his mail box ,it was the foundation of a Social Inbox [2] .
4- Facebook Applications: the biggest motivation and convenient integration between many application and the web site interface of the Facebook pages, which leads many such users to still connected with Facebook home pages and being in touch with many advertisers ,besides the applications are games , this commercial OSN attracts users and advertiser to be there ,the integration between the games components and users, their profiles ,images ,lists of their friends and already joined groups ,increase the functionality and integration levels with different components [10].

Most Facebook Applications are more simplified than most casual modern games, requiring an average of one or two-click actions and supplying a random outcome mostly independent from skills, usually in a very short span of time (seconds). Frequently, the actual gameplay is substituted by a text offering a narration of the events and their outcome, as some sort of prize in exchange of the minimal (one click) engagement required. Facebook Applications feature several elements of social play, Making the participation of the user's Friends is a must, in order to access the Application, or by proposing primers for Confrontation with others [2][10] .

3. FACEBOOK - DISTRIBUTED SYSTEM DESIGN

3.1 Architecture

Facebook, the online social network (OSN) system is relying on globally distributed datacenters which are highly dependent on centralized U.S data centers, in which scalability, availability, openness, reliability and security are the major System requirements. When founded in 2004 it was such a dream to be the largest OSN by the year of 2013 putting the system on the surface of risk unless it well designed and protected against failure and attacks [8]. the architecture of the system ,the scheme here is 3 tier architecture or more (4 tier) ,in which the data folw originated form clients requests that are servedby the follwing steps :

1- Initially by dedicated webservers,these web serveres are highly connected in high available scheme to handle billions of requests and aggregate the logs coming from different webservers .
2- then they are redirected in uncompressed format to the Scribe–Hadoop Clusters they are dedicated for logs aggregations , the later is then communicate the Hive –Hadoop servers cluster ,these servers are divided in two categories ,the Production and the Adhoc ,they are clusters of servers that are balancing acoording to the priority of jobs, for example the Production servers are dedicated to the jobs that being strict in delivery deadlines time constrains , while the adhoc cluster is serving the low priority

batch jobs as well as any ad hoc analysis that the user want to do on historical data sets .

3- federated Mysql is the data base engine which hold the data bases holding up the whole system [8] .these tier parts are described in the figure (2) .

3.2 Distributed systems components:-

Scalability and reliability are mandatory requirements according to the globalization of the system, Facebook is global OSN that serving billions of requests and being responsible for replying back to their requests in just few seconds, and not being too late, these requirements need scalability ability in size, geographically scalability and save the robustness of the system [9]. Systems design, big data processing and analysis and huge Storage that are examples of these components that are Facebook relying on, because of their ability to holding text, multimedia and many third party applications and advertisement and put them on the surface to the users [8] . Facebook is relying on Hadoop platform, which is well suited to deal with unstructured text,logs,and events steams , and structured data, as well as when a data discovery process is needed. it is built for the purpose of handling larger volumes of data, so preparing data and processing it should be cost prohibitive [2][3] .

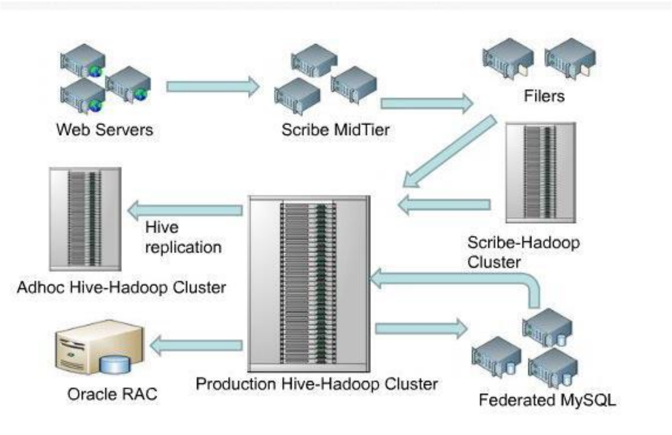


Figure (2): Facebook system architecture

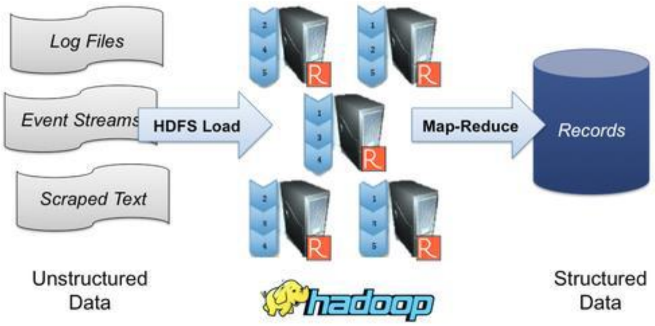


Figure (3): Hadoop system on Facebook

Hadoop has Two main components: -

1. Map–Reduce, which dedicated for Computation. (M-R)
2. Hadoop Distributed File System (HDFS), deals with Storage. See figure (3).

A typical Hadoop environment consists of a master node, and worker nodes with specialized software components. Hadoop consists of multiple master nodes to avoid single point of failure in any environment. The elements of master node are:-

Job Tracker: Job tracker interacts with client applications. It is distributing Map and reducing tasks to particular nodes within a cluster.

Task tracker: it is process receives the tasks from a job tracker in in the master node like Map, Reduce it to specific cluster node and shuffle.

Name node (NN): they are responsible for keeping track for each file in Hadoop Distribute File System HDFS ,a client application contact NN to locate file ,delete ,copy ,or add.

Data Node (DN): they are responsible for storing in HDFS , they are keeping indexes for files stored in , they are interact between client applications and the NN .providing the clients with name of NN that are hold the required data .

Worker Nodes: they are the servers who are responsible for processing tasks; each worker (slave) holds DN and a task tracker. See figure (4).

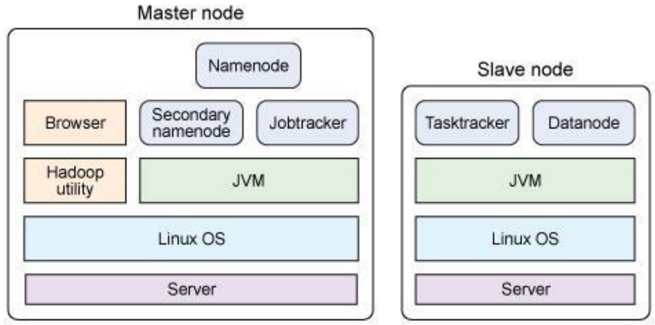


Figure (4) : Hadoop master /slave architecture

3.2.1 Map Reduce (M-R)

Tera Bytes and Peta Bytes of data to get processed and analyzed daily by Facebook data centers. So to handle them we use Map Reducer which basically has two major phases map & reduce they are divided in the following steps:-

1- The Mapper - Reducer uses key/value pairs to index any data comes from HDFS and being divided into 1- The Mapper - Reducer uses key/value pairs to index any data comes from HDFS and being divided into blocks, replicating these values to protect system in case of failure.

2- Submit the M-R Job and its details to the Job tracker that contact the task tracker on each DN that schedule Map Reduce tasks.

3- When Mapper process data blocks and generates a list of key value pairs. Sorting the list of key value pair and transfers mapped results to the reducers in sorted format .

4- M-R merge list of key value pairs to generate final results. Storing in HDFS and replicated ,clients now will be able to read from HDFS easily [3]. The steps are summed up in figure (5).

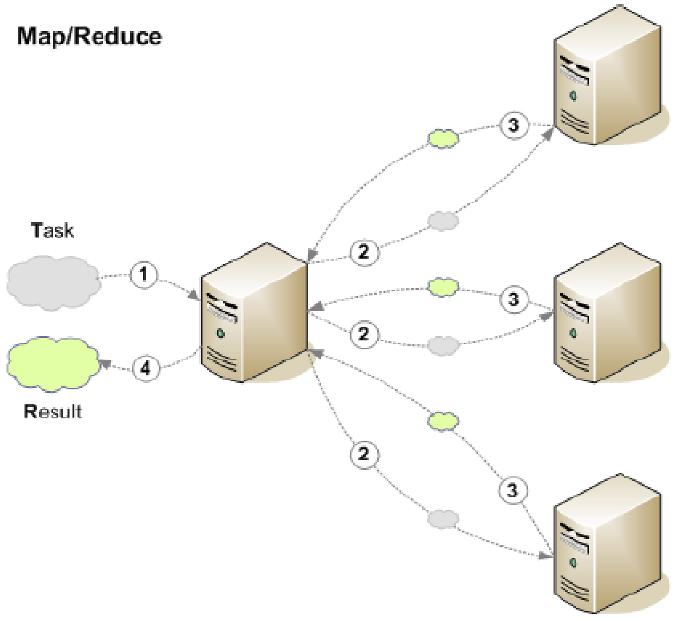


Figure (5): Map /Reduce whole steps

3.2.2 Hadoop distributed file system (HDFS)

Distributed file system that serve the Facebook is mainly Hadoop distributed file system (HDFS) ,which is designed to run on low-cost hardware ,and being highly fault-tolerance (as it supports block replication) . HDFS is designed to store very large data sets reliably; it is able to stream those data sets at high bandwidth to user applications. It used In a large cluster, thousands of servers are directly attached storage and execute user application tasks. By distributing storage and computation across many servers, which give the system ability to dynamically scale ,the resource can grow on demand while remaining economical at every size and retaining the system available and reliable I. An HDFS instance may consist of hundreds or thousands of server machines, each storing part of the file system's data; HDFS is designed more for batch processing rather than 5 interactive use by users. The emphasis is on high throughput of data access rather than low latency of data access, a typical file in HDFS is gigabytes to terabytes in size. HDFS applications need a write-once-read-many access model for files. This assumption simplifies data consistency issues and enables high throughput data access [2]. HDFS exposes a file system namespace and allows user data to be stored in files. Internally, a file is split into one or more blocks and these blocks are stored in a set of Data Nodes The existence of a single Name Node in a cluster greatly simplifies the architecture of the system. The Name Node is the arbitrator and repository for all HDFS metadata. The system is designed in such a way that user data never flows through the Name Node [3][4] ,see figure (6).

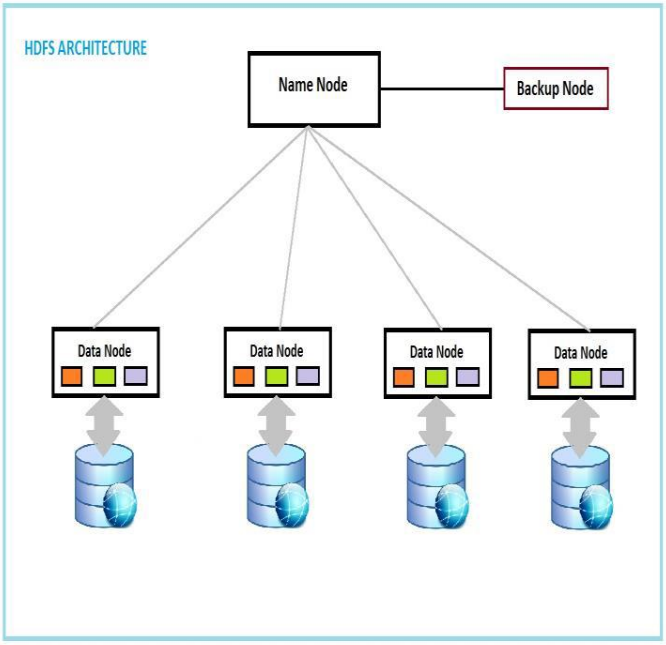


Figure (6): HDFS architecture DN and NN

3.2.3 Hadoop and Hive

In Facebook Hive is a data warehouse infrastructure built on top of Hadoop technology, that provides tools to enable easy data summarization, heavily reporting ,adhoc querying and analysis of large datasets data stored in Hadoop files HDFS . Providing a mechanism to put structure on this data and it also provides a simple query language called HiveQL which is based on SQL and which enables users familiar with SQL to query this data [1]. In System design of Facebook without Hive, the same job would take hours if not days ,in order to move to the second phase and author in map-reduce process . While Using Hive the task could be expressed very easily in a matter of minutes. It has been possible with Hive to bring the immense scalability of map-reduce to the non-engineering users as well – business analysts, product managers and the like who, though familiar with SQL would be in a very strange environment if they were to write map-reduce programs for querying and analyzing data by themselves and without Hive-QL syntax [1]. Figure (7) show Hive system architecture.

3.2.4 Apache HBase

Facebook messaging system has recently added to the application, by the support of Apache HBase which is a database-like layer built on Hadoop designed to support billions of messages per day. The application's requirements for consistency, availability, partition tolerance, data model and scalability. Enhancements made to Hadoop to make it a more effective real time system, Facebook made many tradeoffs while configuring the system, to add significant advantages over the shared MySQL database scheme used in applications at Facebook [2]. HBase will add the following to Facebook as it moves to real time rather than being offline ,this emerging movements are support Facebook billion messages capacity which will be increased with minimal overhead and no down time , with Highly write throughput ,efficient and low-latency that support the strong consistency semantics within a data center, the efficient random reads from disks , and being highly available specially in disaster recovery , and fault isolation ,and retaining the atomic read –modify – write primitives .It added a zero downtime in case of individual data center failure, running on Active-Active serving capabilities across different data centers [2].

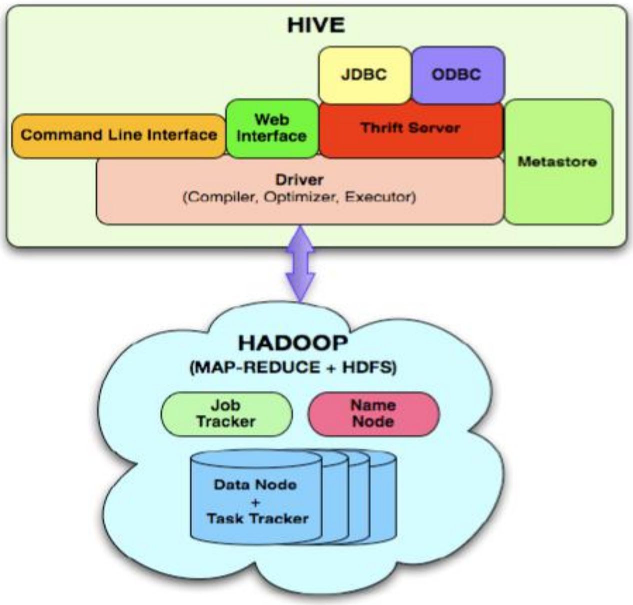


Figure (7): Hive System Architecture

3.3 Communication

3.3.1 Communication in general system

Facebook User's contact the updates by establishing a TCP connection oriented (persistent in case of polling updates), and receives HTML responses post back to them by browsers [9]. Thinking of these traffic generators, and the locations of Facebook datacenters that are centralized in US California : Santa Clara ,Palo Alto ,Ashburn ,the bandwidth and latency measured form outside the U.S users and these distributed datacenters will be risky dangerous ,and definitely encouraged the decision taker to think of multiple solutions to maintain the network reliability and system availability and protect the system from network bottleneck problems [9] . The solution was to let Facebook servers Content Delivery Network CDN handling the objects and well co-located geographically illustrated in figure (8). CDN are spanning widely, and geographically distributed through Russia, Egypt, Sweden, and UK ,etc .

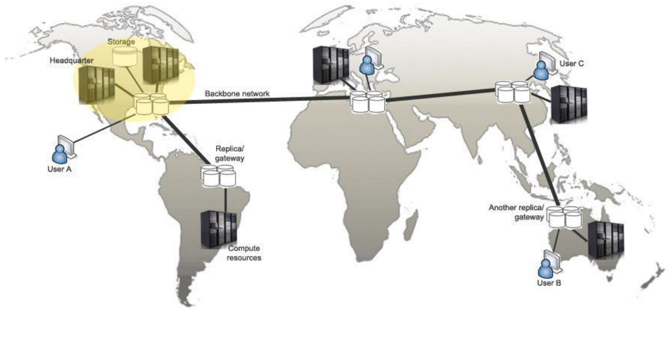


Figure (8): CDN support Facebook network

Although the CDN regional servers posed an attractive solutions for infrastructure expansion another solutions mentioned here will serve a good support for the huge growth and datacenters extensions ; TCP proxies and regional OSN caching servers would be attractive solutions to enhance the network performance and reduce latency; unfortunately these solutions are under tacking and are not being applied yet, which cause slow performance and long latency measurements in Facebook overall statistics [9]. In figure(9) : we can see that a user will contact webservers in U.S ,CDN should maintain connected in more than 4 steps then CDN complete serving the user requests ,while figure (10) which use TCP proxy or figure (11) that illustrated the OSN cache solutions .

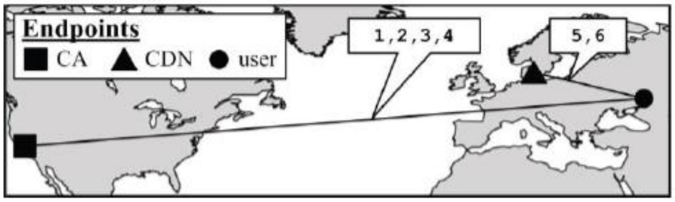


Figure (9): current state for Facebook communication

In TCP proxies figure (10) ,user can be served totally by contacting his regional server ,sometimes there is a need to establish the connection form the original servers and being completed by their CDN , while in OSN cache regional servers in figure (11); the requests are being served totally by them ,sometimes there is a little bit need to be asking the original servers ,these solutions will help Facebook to be away from bad performance ,and increase the capability for the system to scale well in the future [9].

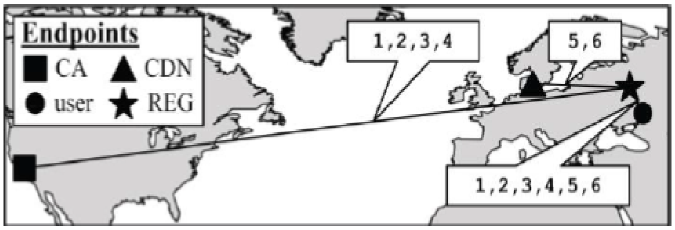


Figure (10): Facebook communication with using TCP proxies.

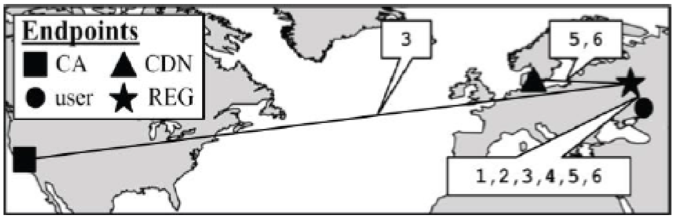


Figure (11): Facebook communication with using OSN cache .

3.3.2 Communication within systems processes

Hadoop servers are compatible with Remote Procedural Call (RPC), in which all coming requests that are redirected from application servers to MY-SQL based architecture servers are served in term of RPC ,this mechanism of communication improved for real time work load since Facebook have published Messaging service in later years of working as online social network ,and being enhance a little bit in Hadoop to be limited with time constrains [2] . Hadoop exploits tcp connections by sending RPCs. When a RPC client detects a tcp-socket timeout limits , it sends a ping to the RPC server instead of declaring RPC timeout .now if the server is still alive and could communicate with clients , client can continue waiting for a response. While in case of a RPC server is experiencing a communication burst, a temporary overhead or load, the client should wait and direct its traffic to the server. And from opposite side in case of throwing a timeout exception or retrying the RPC request causes tasks to fail unnecessarily or add additional load to a RPC server [2]. In another side of system, choosing infinite wait will have an impact on any application that has a real time requirement. For example An HDFS client occasionally makes an RPC to some Data node DN , and it is not good when the Data Node fails to respond back in time and the client is stuck in an RPC. A better scenario is to fail fast and try a different Data Node DN for either reading or writing. Hence, Hadoop has the ability for specifying an RPC-timeout for each request depending on the job which could be served from application servers or want to call data base servers that had to call HDFS in deed .when starting a RPC session with a server; Hadoop is responsible for these tuning and configurations [2][3][4]. Facebook Messaging service combines existing old fashion Facebook messages service with e-mail messaging , chat, and SMS. Hadoop offer a persisting communication between clients, it added a new threading model also requires messages to be stored for each participating user this feature gives user ability to manage his social inbox account with highly write /read throughput ,the idea of this threading model As part of the application server requirements, letting each user be sticky to a single data center at a time [2].

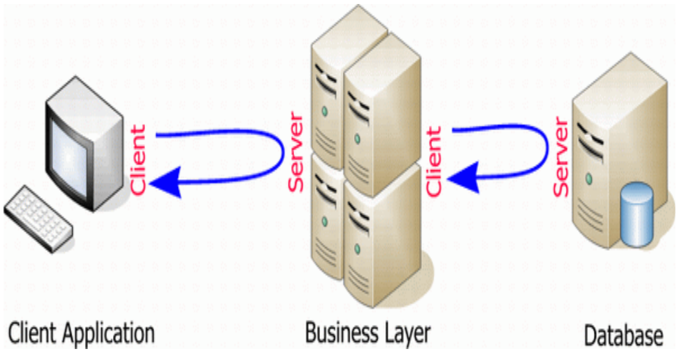


Figure (12): RPC between Hadoop servers

3.4 System design enhancment

In just few years Facebook distributed system has a traditional design, in which Hadoop and Hive were working together to perform tasks for storage and analysis of large data sets .these analysis are classified in to two categories, most of them are offline batch jobs to maximize the throughput and efficiency and the others are online jobs. These workloads are read and write large amount of data form disks sequentially.

3.4.1 Memcahed servers

Recent design of Facebook, let Hadoop performing a random access workloads that provides low latency access to HDFS, by using a combination of large clusters of MySQL databases and caching tiers built using memcached ,that will be support a better in performance while all results from Hadoop are directed to MySQL or memcached for consumption by the web tier side [2] , see figure (13) .

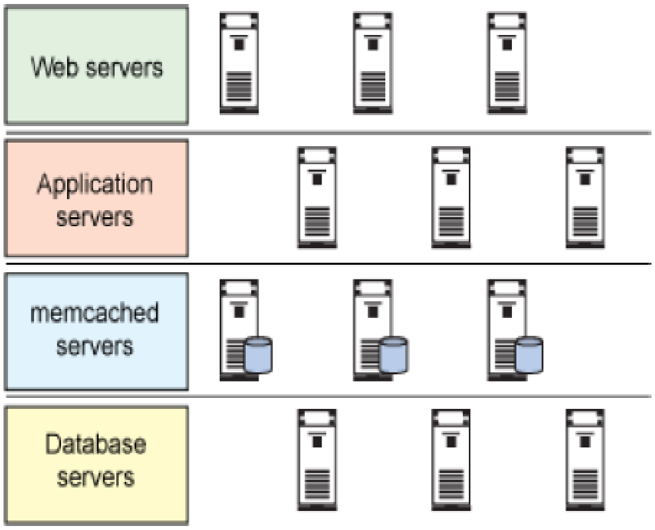
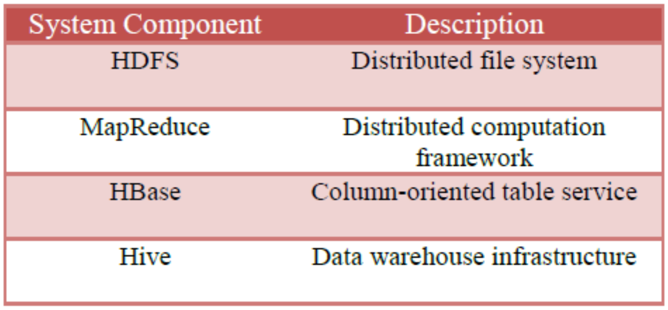


Figure (13) : memcached servers

Recently, a new generation of applications has been applied at Facebook in which requires very high write throughput and cheap and elastic storage, while keeping low latency and disk efficient sequential and random read performance [1][3][4].MySQL storage engines are proven and have very good random read performance, but suffer from low random write throughput. Scaling up Database MySQL clusters rapidly is difficult to deal with, because of the needs to maintain load balancing and have long and high uptime. Administration of MySQL clusters requires higher managing overhead and costly hardware [2] [3] [8]. We sum-up the whole system components in the figure (14) below and listing the major parts that we have discussed in this paper in table (1).



System Component	Description
HDFS	Distributed file system
MapReduce	Distributed computation framework
HBase	Column-oriented table service
Hive	Data warehouse infrastructure

Table (1): Hadoop project components

ISSN 2347-4289

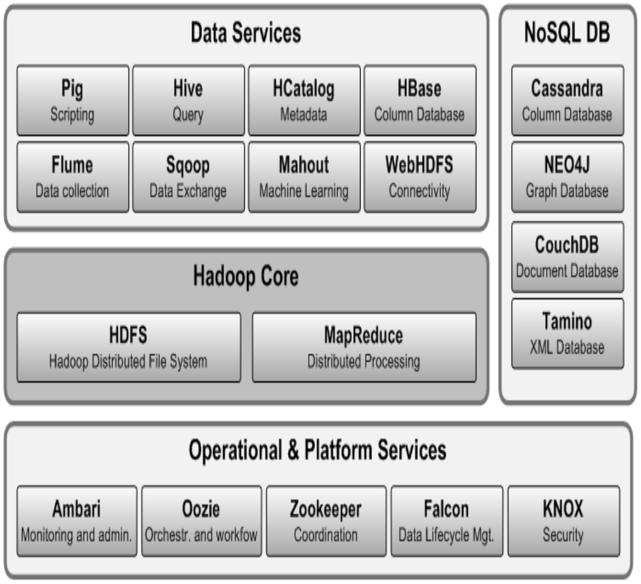


Figure (14) : whole recently system architecture

3.4.2 Colud computing support

Redundant Cluster servers are used to hold the whole system in Facebook, now the system in consist of physical server that needed to be extended day by day, this scheme of datacenters hosted Facebook servers is subjected to be at risk one day and subjective to many problems being as limitations for growth [5][6]. Nowadays cloud computing is offering a powerful environment to scale web applications without difficulty. Using such schemes of resources on-demand for many scaling points as web applications, storages, and servers Cloud computing aims to deliver services over the network it provides ability to add capacity as needed ,it is basically use virtualization techniques to turn computer resources in to virtual guest depending on availability of such resources in the hosting environment ,guest computers are running sharing the same resources while they are isolated in their design and configurations ,while cloud computing offer accessibility to the users form anywhere though their connected devices to their published applications ,many trends appear here to save the data navigated between users and applications in secure manner [6]. This shift for the technology will put data centers and their administrators at the center of distributed network, as computational power, web applications, resources that being shared among them ,bandwidth and storages are all managed remotely. While Facebook datacenters until now is physically hosting all its servers and data bases in real data centers, and not depend on cloud computing to scale its platform or infrastructure; cloud computing such application as a service will be a good example to exploit the scalability gain for virtualization technology to meet some demand on growing requests and numerous traffic and offering a lot of increasing demand appear to integrate many applications with Facebook application system [5]. While scalability is a measure of ability of an application to expand to meet enterprise business needs ,resources under demand are anything could be required or shared by the system users ,it is ranging from processor, storage space and network bandwidth ,these resources will affect primary the system performance and degrade the application behavior when they are running on shortage of them ,when application is not scale well it is encounter the performance and service availability as demand increase [6]. Scaling indicators should be determined well in order to tune applications regarding these indicators ,such : number of concurrent users (they are access in the same time) ,number of active connections being served ,number of requests per seconds ,and average response times per request ,sampling of these indicators in real time , based on historical values used and some of predictable ones are set ,resulting in scaling up or down decisions are being taken for web application instances ,this is being done by let the amount of web servers and web application component to grow or shrink upon demand this is dynamic scaling feature [5][6] . see figure (14) .

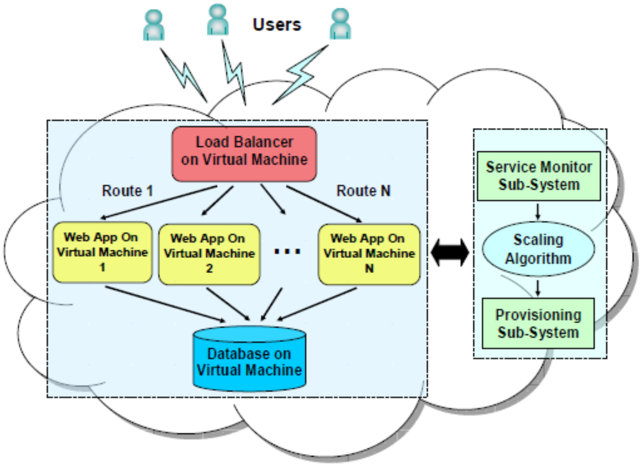
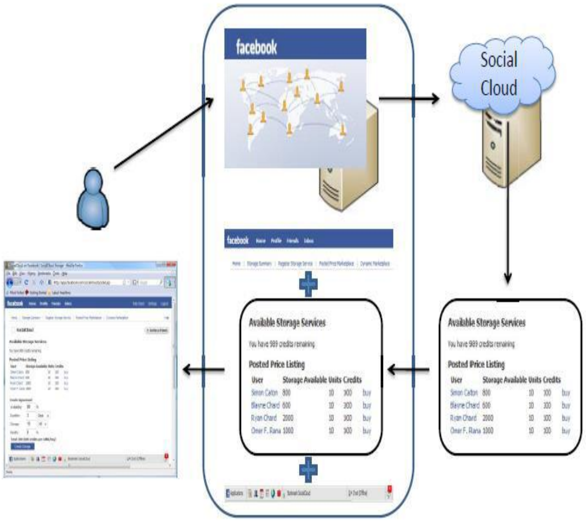


Figure (15): Architecture to scale web applications in a Cloud

3.4.3 social network as virtual organization

The structure of social network is essentially a dynamic virtual organization, in which a trusting relationship is inherently among friends relationship, while resources (information ,hardware, services) are shared among these social network , a social cloud which offer a low level abstractions of computations and storage ,could easily acts as a complementary building block for any social network ,this is because a social cloud is a scalable computing model in which virtualized resources are shared by users and dynamically provisioned among them, some service level agreement (SLA) should be exist to manage the sharing process of virtual resources. Cloud here offer the scheme of application as a service APAS [6]. Cloud platforms are used to host social networks or to create such scalable applications PAS, Facebook applications is such example and a particular part that play significant role in the social clouds, these applications exploit Facebook methods in order to render friends ,events, relationships, groups, profile information ,and multimedia as audio /video, and Facebook markup language (FBML),these range of data enable completely integration between Facebook components and these applications ,which are definitely are not hosted within Facebook environment they are hosted independently [6] . All communications between specific user and these applications are done isolated without interrupting Facebook servers ,which is more attractive performance behavior, since once a user request an URL for any application, and all communications later are served from specific application server hold that application ,this scheme is adding a positive point in design considerations ,Facebook JavaScript (FBJS) are used often to request Facebook servers asynchronously and in transparent manner without routing through applications servers [5][6].



Available Storage Services

You have 989 credits remaining

Posted Price Listing

User	Storage Available	Units	Credits	
Simon Caton	800	10	300	buy
Blayne Chard	600	10	300	buy
Ryan Chard	2000	10	300	buy
Omer F. Rana	1000	10	300	buy

Available Storage Services

You have 989 credits remaining

Posted Price Listing

User	Storage Available	Units	Credits	
Simon Caton	800	10	100	buy
Blayne Chard	600	10	100	buy
Ryan Chard	2000	10	100	buy
Omer F. Rana	1000	10	100	buy

Figure (16): Facebook applications hosting environment

The Social Cloud utilize web services to create scalable, distributed and decentralized infrastructure, with storage as a service that complete the scenario well done, each storage service is relying on a web application to deliver content to the Facebook application with no need to route the requests through the social cloud applications, this earlier steps done by using JavaScript JS and dynamic AJAX invocations [6]. Users easily can create a storage by passing agreement to the storage service ,they access their virtual storage and create their own resources , keeping track for their storage contents ,view storage limits and used/available spaces ,managing files and folders that the storage holds ,and getting agreement outlines and subscribing information [5][6].see figure (16)

4. CONCLUSION AND CRITICAL EVLUATION

We have explored Facebook as a case study for distributed sytem,discussed the system features and providing a detailed system design architecture, communications and system components .this paper is provide an extensive study for Facebook distributed system inside its data center The system is built on top of highly equipped data centers that are provide the system the availability and reliability ,the Hadoop project is an example of this system that Facebook in built on top of its technology . Using the clusters for the data base systems, load balancing webservers and application servers that are responsible for replying on user’s requests, the ability to compress the traffic between servers to save the bandwidth and the isolation between jobs that are derived from user’s requests. Being geographically distributed by using centralized data centers located on US and being replicated by distributed CDN, is providing the system the level of acceptable scalability, with the CDN the system is still working in an acceptable levels, the TCP proxies and OSN cache servers will provide the system the up limits scalability they are under studying and research and unfortunately are not applied yet. Hadoop projects and whole components are example of success story that provide Facebook system with its requirements to be the most popular social network by the year of 2013 ,while rapidly added services and being occasionally updating their services ; messaging and chat are examples of these services that requires Hadoop to do a little bit enhancements on their design to be real-time system rather than to work offline processing and save the low latency issues required to access the HDFS as fast as possible , adding RPC timeout as final enhancement . Memcahed severs are also another example of these enhancements to decrease the load of accessing the data base in each case that require access to the data base. Cloud computing is model example that Facebook used to integrate with its features and services .this integration is done without any infrastructure modifications or any architectural changes , this is because cloud computing is offering an acceptable solution for integrating Facebook with such examples of cloud applications .the most interesting examples of these solutions the social cloud being built by the virtualization organizations that provided ,these are being scaled dynamically and on demand .

5. REFERENCES

[1]. Thusoo, Ashish, et al. "Hive-a petabyte scale data warehouse using hadoop." Data Engineering (ICDE), 2010 IEEE 26th International Conference on. IEEE, 2010.

[2]. Borthakur, Dhruba, et al. "Apache Hadoop goes realtime at Facebook." Proceedings of the 2011 ACM SIGMOD International Conference on Management of data. ACM, 2011.

[3]. Shvachko, Konstantin, et al. "The hadoop distributed file system." Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on. IEEE, 2010.

[4]. Lakshman, Avinash, and Prashant Malik. "Cassandra: a decentralized structured storage system." ACM SIGOPS Operating Systems Review 44.2 (2010): 35-40.

[5]. Chard, Kyle, et al. "Social cloud: Cloud computing in social networks." Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on. IEEE, 2010.

[6]. Chieu, Trieu C., et al. "Dynamic scaling of web applications in a virtualized cloud computing environment." e-Business Engineering, 2009. ICEBE'09. IEEE International Conference on. IEEE, 2009.

[7]. Yang, Bo-Wen, et al. "Cloud Computing Architecture for Social Computing-A Comparison Study of Facebook and Google." Advances in Social Networks Analysis and Mining (ASONAM), 2011 International Conference on. IEEE, 2011.

[8]. Thusoo, Ashish, et al. "Data warehousing and analytics infrastructure at facebook." Proceedings of the 2010 ACM SIGMOD International Conference on Management of data. ACM, 2010.

[9]. Wittie, Mike P., et al. "Exploiting locality of interest in online social networks." Proceedings of the 6th International COnference. ACM, 2010.

[10]. Rao, Valentina. "Facebook Applications and playful mood: the construction of Facebook as a third place." Proceedings of the 12th international conference on Entertainment and media in the ubiquitous era. ACM, 2008.