CHAPTER 2

LITERATURE SURVEY

2.1 WEB APPLICATIONS PERFORMANCE STUDIES

Marco Scarpa et al (2004) explained a modeling technique for the performance analysis. A procedural approach for the performance analysis of web-based and web enabled searching applications on the Internet. It puts tremendous amount of effort in investigating the behavior of the Client/Server (C/S), Remote-Evaluation (REV) and Mobile-Agent (MA) for successful establishment of correspondence paradigms and describes how Petri-net models can be developed to realize performance index attributed to the designer to enhance the efficiency of his distributed applications. This is to unravel and identifying a set of models that can help to understand the environmental situations in which such paradigms should be preferred or combined for the sake of improving ultimate performances of a disseminated system. In particular, the Marco Scarpa et al been sticking to the point of mooting a modeling practice applied to an Information Reclamation application on the World Wide Web. An analytical evaluation through the solution of non-Markovian Petri-net models is provided, which entitles the scholar to pick out the main parameters, as well as the way they communicate with the rest of the systems, to be taken into consideration when distributed applications are to be designed. An experimental environment is also studied in the wake of obtaining the real measurements used to validate the analytical models and also it has played a level playing role.
Peng Cheng Xiong et al (2010) have given a transaction on systems, Man and Cybernetics Part-A systems and humans. With the development of enterprise-wide and cross-enterprise application integration and interoperation toward web service, web service providers would not restrict themselves only a particular portion of area for their consideration, but also try to fulfill the radicalized requirements of web service users but also satisfy their nonfunctional conditions in order to make or pretend to make it as surviving one in the lucrative competitive market. A hot research topic is how to put together web services to meet their genuine demand given the diversifying nature of user requirements, peculiarity of rendering service modules’ performance, and limitation of available resources are used to be taken into account. A web service configuration net based on Petri nets in order to put on display the web service putting together the pieces of thing in a formal way. Then, a possible algorithm is being presented to help choose the best combination of highest quality of service to meet users’ nonfunctional and casual requirements.

Michael Grottke et al (2006) have done an outstanding analysis on web server. This is also called research choreographed the developmental aspect of resource utilization in a web server while subjecting it to an artificial intelligence workload. Authors first of all used to mobilize the data from several system gadgets associated with it and the usage and activity parameters. Non-parametric statistical methods are then functional toward detecting and estimating trends in the data sets. Finally, authors’ fits time series models to the data collected. Unlike the models used previously in the research on software aging, these time series models allow for seasonal patterns, and authors’ show how the exploitation of the seasonal variation can help in adequately predicting the future resource usage. Based on the models employed here proactive management techniques like software upgradation
that going to be the instrumental and triggered by actual measurements can be build around it.

Swaminathan et al (2007) explained caching and replication strategies for web applications. The onus was on with this discusses an avian view of various measurable methodologies that has to be put side by side to have a comparison for analyze their features and performance. The slowing down of web application will be attributed to number of reasons, but the most well speculated one was the dynamic generation of web documents. There are some real time modern web sites such as Amazon.com and Slashdot.org were don’t deliver fixed of static of fixed paged which cannot be altered the pages, they tend to generate content on the basis of fly each time they receive a request, has played a crucial role in customizing their pages for each user. Clearly, generating a web page in response to every request takes more time than simply fetching static HTML pages from a server. Whenever generating a web page on dynamic generation usually warrants issuing more than one query to a database which is at present in usage. Hence the access time to the database could easily go out of the control of the entities hand when the request load is high. Tycoon of industries as well as academic scholars has developed several techniques to overcome this problem. The down-to-earth and uncompromising one is web page caching, in which divisions of the HTML pages the application generates are cached to serve future course of actions. One more debutant that is Content Delivery Networks (CDNs) such as Akamai was perceived to have the capacity to do by deploying edge servers around the internet to locally cache web pages and then deliver them to clients. This can be assumed by dispatching pages from edge servers located in the proximity to the node or the clients, CDNs (Content Delivery Network) reduce each request’s network waiting time. Page-grapping or catching technique works well if the same cached HTML page is equipped with the capability to answer many requests to a particular web site or web
application. The above procedures were proved to be correct one as well as reliable one, but with the growing drive towards personalized web content, generated pages tend to be unique and robust for each user, so it will trim down the benefits of page caching techniques. The idea of page caching’s limitations has mooted and triggered the CDN and database research community to investigate new approaches for scalable web application hosting. By bi-furcating these approaches broadly into four techniques, like application code repetition, cache database records, cache query results, and entire database replica can be achieved. In spite of numerous research efforts, it is concentrated on these approaches, very few works have analyzed their merits and demerits examined their performance.

Kanodia (2003) ensured latency targets in multiclass web servers. Two recent advances have resulted in significant improvements in web server quality-of-service. First, both centralized and distributed web servers can provide isolation among service classes by fairly distributing system resources. Second, session admission control can protect classes from performance degradation due to overload. The goal is to design a general "front-end" algorithm that uses these two building blocks to support a new Web service model, namely, multiclass services which control response latencies to within pre specified targets. Author’s key technique is to devise a general service abstraction to adaptively control not only the latency of a particular class, but also to bind the interclass relationships. In this way, author captures the extent to which classes are isolated or share system resources (as determined by the server architecture and system internals) and hence their effects on each other's QOS. For example, if the server provides class isolation (i.e., a minimum fraction of system resources independent of other classes), yet also allows a class to utilize unused resources from other classes, the algorithm infers and exploits this behavior, without an explicit low level model of the server. Thus, as new functionalities are incorporated
into web servers, the approach naturally exploits their properties to efficiently satisfy the classes’ performance targets.

Huan Yang (2011) with the help of real-valued negative selection algorithm it was easy for us to evaluate the age of the particular web server. There is no doubt that the highly used web servers are vulnerable to fall in the pray predator that is ageing. Ultimately those things lead to the degradation of web server of soaring in the rate of failure of the web server performance. In order to avoid this scenario detection of web server aging assumes an important role in web server management. The methodology and the technologies that are followed so far succeeded to some extent, for instances taking aging samples of the given web server for analyzing the age and the performance of the web server, but failed to detect previously unseen aging because of the nondeterministic nature of the web server aging. In this paper, an immune-inspired real-valued negative selection algorithm, which needs normal samples to perform anomaly detection, is proposed to distinguish web server aging (non-self) from normal state (self). The basis of the detection is aging that causes exhaustion of system resources and performance degradation, namely performance deviation from the normal state. Since normal samples are much easier to acquire from running web server than aging samples, this algorithm will improve the web server aging detection performance, especially of unknown ones.

2.2 QUERY OPTIMIZATION STUDIES

Dandan Li et al (2010) have optimized queries in Relational Database System. Oracle provides more than a few options to give support to presentation, such as partitioning outsized tables, by means of materialized views, storing plan outlines, and scores of others. This subdivision examines how DBAs can transport into play these techniques to give moral and logical
support to developers’ hard work to enhance the competence of their submission code.

Ordonez et al (2010) have optimized Linear Recursive Queries in SQL. In this thesis, present SQL implementations for two deep-seated or basic algorithms: half adolescent and direct. Five query optimizations are studied: 1) storage and indexing; 2) early selection; 3) early assessment of non recursive joins; 4) pushing duplicate elimination; and 5) pushing aggregation.

Feng dong Sun (2011) described attributed to the value addition by way making enormous amount of pressure mounted paging query optimization of enormous amount of data in oracle 10g database. In this piece of writing Paging query effectiveness of gigantic data directly relates in the direction of the carrying out effectiveness of appliance organization in oracle database. Presented an optimized solution for paging query in oracle 10g database, which includes many key technologies, such as database optimization, SQL optimization, cursor variables, batch binding, dynamic SQL and so on. A shared, optimized stored procedure is issued. Compared with the traditional methods, the paging query efficiency of massive data improved obviously.

Tan Bao et al (2010) have presented a method for performance optimization based on stored procedure in RDBS project. When system encounters large amount of data, complex transaction rule and high system security required in relationship database system project, the efficiency, maintainability, data integrity and reliability are eagerly to be solved. Using stored procedures can improve the questions mentioned above. By taking Query Service System of Teaching Affairs as example, this paper analyzes the meaning of stored procedure in database and key problem of application strategy, illustrates query optimization method based on stored procedure, and also presents some other approaches to improve the performance of RDBS.
Finally, the performance evaluating result before and after the optimization is provided, which shows the method is practical and effective.

2.3 DATA TRANSMISSION STUDIES

Kaiyu Wang (2009) evaluated the performance of J2EE web applications with Queuing Networks. The prediction of systems response time for different intensity of users’ requests, esp. identifying requests loads causing performance degradation becomes an important issue in the design of an e-commerce system. A method for modeling application server environment by the use of queuing networks has been proposed. The physical components of the environment are represented as service centers which form a separable queuing network. Model parameters, i.e. average service times and average visit counts can be experimentally measured in a test environment. Model outputs are response times for different request classes as well as other parameters, describing systems performance. Tools supporting the measurements and model calculations have also been developed. Experiments carried out on a J2EE web application show that the model predicts real system’s performance with satisfactory accuracy.

JunweiGe (2010) analyzed sever performance degradation based on average load chaotic time series forecast. A long-running web software system may lead to the exhaustion of resources, which cause performance degradation. To solve that problem, it needs to predict the crucial resources using situation, and then carry out the proper software rejuvenation strategies. This identifies the average load chaotic character which can be described by using G-P algorithm to analyze correlation dimension changing with embedding dimension, then get the largest Lyapunov exponent through small data method and build chaotic time series prediction model based on largest Lyapunov exponent for average load time series. The trial and error method the author has been using shows that the
forecast form can in particular make short-time forecast to the web server’s load, which can efficiently estimate the performance degradation situation and provide groundwork for the software renovation.

Mohamed Elsalih Mahmoud and Xuemin Sherman (2010) offered a Practical Incentive System for Multihop Wireless Networks. An enticement system is proposed to encourage the nodes’ cooperation in MWNs. The payment model has been urbanized to implement micropayment for cooperation stimulation efficiently. Reducing the operating cost of the payment receipts is necessary for the practical implementation of an inducement system due to the soaring frequency of low-value transactions. Therefore, one fixed-size receipt is generated per session, regardless of the packets’ number. Attaching the hash of the signatures as a replacement for of the signatures can reduce the receipt magnitude significantly, and the receipt-aggregation procedure has been used to generate a smaller size receipt for compound sessions. In accumulation, to the knee-jerk receipt submission apparatus has been anticipated to trim down the number of submitted receipts and protect against collision attacks by a less quantum of number of redundant receipts with restricting the crashes’ gains probabilistically. Our understanding and model prototype demonstrate that the anticipated the incentive system can secure the payment and significantly trim down the receipts’ accumulation sphere and the number of generated and submitted receipts.

Dominik Seiler et al (2011) were proposed an approach for Data Transmission between Multimedia -Web Services which addresses the issues pertaining to web services are identified also known the SOP with AOP.
2.4 LOAD BALANCING STUDIES

Zilio et al (2006) have devised and envisages everyone that the Load Balancing for Multi-tiered Database Systems. A materialized observation or MQT is an secondary chart with pre computed data that can be used to extensively perk up the presentation of a database query. A Materialized Query Table Advisor (MQTA) is time and again used to advocate, generate MQTs. Modern technology MQTA works in an individual database server where MQTs are placed on the same server as that in which the base tables are located. The MQTA does not apply to a federated or degree out scenario in which MQTs need to be placed on other servers close to applications (i.e. a frontend database server) for offloading the workload on the backend database server. In this paper, author proposed a Data Placement Advisor (DPA) and load balancing strategies for multi-tiered database systems. Built on top of the MQTA, DPA recommends MQTs and advises placement strategies for minimizing the response time for a query workload. To demonstrate the benefit of the data placement advising, we implemented a prototype of DPA that works with the MQTA in the IBM® DB2® Universal Database™ (DB2 UDB) and the IBM Web Sphere® Information Integrator (Web Sphere II). The evaluation results showed the level of erecting substantial amount of improvements of workload response times when MQTs are cleverly suggested and positioned at a front end database server subject matter to breathing space and load distinctiveness for TPC-H and OLAP type workloads.

Chiang Lee and Hua (2000) made everyone who came across the dynamic load balancing, informed about the functionality of the multicompiler database. Moreover the multiprocessor known to be shared nothing multiprocessor architecture is known to be more scalable to support very large databases. The strategic comparison of remaining strategies’ a
hash-based join algorithm is particularly efficient and easily parallelized for this computation model. Even though the hardware architecture being perceived to be very sensitive to the skew distribution. Unless and until parallel hash join algorithm includes some dynamic load balancing algorithm it has to face the counter effect of the load balancing system. And also it has to face the performance degradation. In this context, author observes that issues related to three parallel hash join algorithms are presented before parallel dynamic load balancing. The author implements a simulator to study the efficacy of these schemes. The simulation model is validated by comparing the simulation results to those produced by the actual implementation of the algorithms running on a multiprocessor system. Our performance study indicates that a naive approach is not able to provide tangible savings. However, the carefully designed strategies can offer substantial improvement over conventional techniques for a wide range of skew conditions.

Dong Li et al (2000) showcased a vibrant load balancing algorithm based on distributed database system. The thesis launch a dynamic load balancing algorithm based on diversified database system. The perception is concern and thought is to facilitate a reasonable request respondent time and transaction throughput velocity in the place of distributing the whole workload among all servers equally and the performance of the algorithm is given at the end of the paper.

Zenon Chaczko and Venkatesh Mahadevan (2011) envisaged the Load Balancing in Cloud Computing. Now-a-days the cloud computing have been facing a serious obstacle is the availability of cloud systems. Terminology can be replaced with the existence is primarily evaluated by ubiquity of information comparing with resource or the source of supply scale. In cloud, load balancing, one of the methods was applied across the
board in different parts of data centers to make sure that the network availability by trimming down the use of computer hardware, software failures and mitigating recourse limitations. This work discusses the load balancing in cloud computing and then demonstrates a case study of system availability based on a typical Hospital Database Management solution.

Brighten et al (2004) had portrayed Load Balancing in Dynamic Structured P2P Systems. Most P2P systems that give a DHT abstraction distribute objects indiscriminately among “the nodes which are all seem to look closely nodes” the way outcome of the fallout in some nodes having $\Theta(\log N)$ times as many objects as the average node. Further one-sidedness of disproportion may lead to non homogeneous distribution of bits and pieces in the identifier breathing space and a high degree of heterogeneity in object loads and node capacities. Furthermore a node’s load may vary greatly over time since the system can be expected to experience continuous insertions and deletions of objects, out of true object influx of new patterns, and uninterrupted influx and going away of nodes. In this thesis, we have mooted to propose an algorithm meant for load balancing in such heterogeneous, dynamic P2P systems. Our mock-up drill results indicates that in the face of swift and going away of objects of widely unreliable load, the system utilization can be achieved by and load balancing for utilization through an opt algorithm for system utilizations as high as 90% while moving only about 8% of the load that arrives into the system. Like that, vibrant system where nodes getting in and going out, our algorithm moves less than 60% of the load the underlying DHT moves due to node arrivals and departures. Finally, the author shows that our scattered algorithm performs only negligibly not as good as than alike federal algorithm, and that node heterogeneity helps, not hurts, the scalability of our algorithm.
Jiani Guo and Laxmi Narayan Bhuyan (2006) expressed Load Balancing for the group-Based Web Server for Multimedia Applications. Authors accept as true that the bunch-based or the cluster based multimedia web server that with enthusiasm generates take into custody units to persuade and keep happy the bit rate and bandwidth necessities of a variety of clients. The media server partitions the work into a number of responsibilities and schedules them on the backend computing nodes for dealing out. As far as stream-based applications, the main blueprint criterion of the arrangement are to make light of the full amount of processing moment in time and sustain the array of media units for each out bound stream. In this thesis, authors’ first plan, put into practice, and weigh up three scheduling algorithms, First Fit (FF), Stream-based Mapping (SM), and Adaptive Load Sharing (ALS), for multimedia transposing in a cluster upbringing. Authors strong-minded in determining that the necessary to envisage the CPU load for each and every multimedia mission and plan them in view of that due to the inconsistency of the individual jobs/tasks. Therefore, proposed an online forecast algorithm that can vigorously foresee the processing time per individual task (media unit). They then proposed two new load scheduling algorithms, namely, Prediction-based Least Load First (P-LLF) and Prediction-based Adaptive Partitioning (P-AP), which can make use of the forecast to improve the presentation. The performance can be evaluated in terms of system throughput, out-of-order rate of departing media streams, and load balancing visual projection through real measurements using a bunch of computers. The performance of the new load balancing algorithms is compared with all other load balancing schemes to show that P-AP greatly reduces the delay jitter and achieves high throughput for a variety of workloads in a varied cluster. It strikes a good balance between the throughput and output order of the processed media units.
2.5 NETWORK SECURITY STUDIES

Al-Shaer and Hamed (2004) unearthed policy paralysis in distributed atmosphere of routing. In this paper, the 2ACK scheme that serves as an add-on mechanism for routing schemes to detect routing misconduct or any non compliance of schemes and to make it less aggressive in terms of negative impact. The main idea of the 2ACK scheme is to send two-hop confirmation message in the form of packets in anti clock direction of the routing path. Ultimately to ease the additional routing overhead, only a small quantum of received data in the form of packets is acknowledged in the 2ACK scheme. Analytical and simulation results are entrusted with responsibility to evaluate the performance of the proposed scheme and the entire onus is on with 2ACK scheme.

A robust novel firewall management toolkit, explained the firewall configuration management and it’s designed by Bartal et al (2004).

Uribe et al (2004) presented a unilaterally decided analysis of without giving up its sovereignty to the effect of decision making before any scheme of firewall and network intrusion sensing system configurations. Exploring proposes of modeling environmental disharmony in order to efficiently and accurately simulate wireless packet delivery. Authors measured the patches of noise traces strain in many different environments and recommended three foremost algorithms to simulate noise from these traces as prototype. For the full-fledged evaluate applying these algorithms to signal-to-noise curves in comparison to existing simulation approaches used in EmStar, TOSSIM, and ns2. The measure simulation accuracy using the Kantorovich-Wasserstein distance on conditional packet delivery functions. Also demonstrate that using a Closest-fit Pattern Matching (CPM) noise model can capture complex temporal dynamics which existing approaches do not posses. To have good links the onus is on with increasing packet
simulation fidelity by a factor of 2; to have a bad link the pocket simulation used is factor of 1.5 for bad links, and a factor of 5 for intermediate links. As our models are derived from real-world traces, they can be generated for many different environments.

Yuan et al (2006) armed with a set of tools for firewall modeling and analysis. Several defying elements detection and reputation systems have been proposed for mobile real time ad-hoc networks, fully relying on direct network observation functionalities, so-called watchdogs. While these approaches are being evaluated in simulations and restricted to selfish packet dropping, the people of our nature is looking only for capabilities of a watchdog detection component in a real network. Authors presented test-bed implementation of defying elements detection and misbehavior detection. Once evaluate both the feasibility and detect ability of attacks on routing and forwarding in the Dynamic Source Routing (DSR) protocol.

Liu and Gouda (2005) explained a nook and corner analyzed redundancy detection in firewalls. Liu and Gouda discusses new reliable redundancy detection that it is equipped with bearing inconvenience of arbitrary faults, including Byzantine faults. This protocol is perceived using a novel and pioneer way of nurturing secure protocols which is based on a well-founded hybrid failure model. Nonetheless our claim of arbitrary failure buoyancy, the protocol need not necessarily invites the cost of "Byzantine agreement", in number of participants and round/message complexity. It can rely on the survival of a simple distributed security kernel-the TTCB-where the participants only execute decisive parts of the code of behavior, protocol operation, under the protection regime of a collapse failure model. Otherwise, participants follow an uninformed failure model. The TTCB provides only a few basic services, which allow our set of rules or protocol to have efficiency
similar to that of accidental fault-lenient protocols: for faults, our protocol requires $f+2$ processes, instead of $3f+1$ in Byzantine system. Besides, the TTCB (which is synchronous) allows secure operation of timed protocols, despite the capricious time behavior of the environment (possibly due to attacks on timing assumptions).

Zubair Fadlullah and Nei Kato (2010) studied the attacks on encoded set of rules that they used to call it as protocols. Zubair Fadlullah and Nei Kato addressed the online detection of attacks against application-level protocols, which are encapsulated inside encrypted sessions. Experiments carried out in the real data network have provided substantial amount of evidence that implementation of the proposed DTRAB in the Monitoring Stub (MS) is feasible. DTRAB is autonomous at the MS that carries out the detection, i.e., the detection method does not need information from other MSs. Furthermore, the MS builds the database portraying the normal protocol behavior profile, which is not dependent on the traffic volume. As a result of this design, the proposed detection scheme manages to avoid false alarms during flash crowd. The conducted simulations demonstrate the effectiveness of the detection technique. Their investigations have considered the attack detection delay and the “failed session detection error rate.” Also addressed, the problem of tracing back attackers against encrypted protocols based on the correlated attack features at neighboring monitoring stubs. As an approach of responding to the detected attacks, this work may be extended to selectively slow down the protocol response as long as the Cusum sequence exhibits anomalous behavior. Admittedly, when IPSEC protocol is employed by end-hosts through a secure tunnel, the transport layer headers may be encrypted and not visible to the MSs. Our future extensions to this work will consider how DTRAB may overcome such issues.
2.6 KEY CHALLENGES IN WEB APPLICATION PERFORMANCE

It is observed that there are many issues that degrade the web application performance in which the following portions are the primary issues.

The major hurdle when accessing the Web applications over the network is higher network traffic due to poor performance. Users are facing this performance issues when accessing the web applications via URL which were hosted on a web server. Traversal from the user-end can take numerous gateways, hubs and proxies in order to access the web server.

Web application performance behaves poorly due to the surplus congestion on database server. Congestion on database server due to more OLTP and OLAP related a task that leads to poor performance of web applications.

Even though web architecture has the adequate network bandwidth and resources, still, web applications facing the situation of getting performance issues while encryption and decryption during data transmission and reception.