INTEGRATING WIRELESS SENSOR NETWORKS WITH CLOUD COMPUTING AND EMERGING IT PLATFORMS USING MIDDLEWARE SERVICES

Ms. Vennila Santhanam [1], Mr. D.B. Shanmugam [2]

[1]M.Phil. Research Scholar, Department of Computer Science, Dr. MGR Chockalingam Arts College, Arni, TamilNadu, India.
[2]Associate Professor, Department of Computer Science, Dr. M.G.R. Chockalingam Arts College, Arni, TamilNadu, India.

Abstract - *Cloud Computing is the new paradigm for internet-based software systems. It provides scalable processing power and several kinds of connectable services. Cloud architecture has many similarities with wireless sensor network, where the nodes do the task of sensing and local pre-processing. The nodes are interconnected with wireless connections. The platform and infrastructure over the internet is provided by Cloud computing applications. We present a model, which combines the concept of wireless sensor networks with the cloud computing paradigm, and shows how the combination will be advantageous for both. Merging these two technology helps in easy management of remotely connected sensor nodes and the data generated by these sensor nodes. For security and easy access of data, cloud computing is widely used in distributed/mobile computing environment.*

Data from different location is sensed by different sensors in wireless sensor network and are uploaded into cloud for getting good storage and transfer the well processed data to the physical world in an efficient manner. The proposed solution would provide data compatibility, bandwidth management, security and connectivity. We need an intermediate layer between the WSN and Cloud called as the middleware. We use the Networked Control System (NCS) that utilizes heterogeneous wireless networks to collect data; cloud services to provide additional computational capabilities and provides information for different types of end users. The proposed System for cloud computing environments is predicated on deportment of Load balancing using honey bee foraging strategy. This algorithm not only balances the load, but takes into consideration the priorities of tasks that have been abstracted from heavily loaded Virtual Machines. The tasks abstracted from these VMs are treated as honey bees, which are the information updated globally.

Key Words: Cloud Computing, Wireless Sensor Networks, Middleware, Networked Control System, Honey Bee Foraging, and Virtual Machines.

1. INTRODUCTION

1.1 CLOUD COMPUTING

Cloud computing is a technology that uses the internet and central remote servers to maintain data and applications. A simple example of cloud computing is Yahoo email or Gmail etc. A style of computing where massively scalable (and elastic) IT-related capabilities are provided "as a service" to external customers using Internet technologies.

Cloud services include online file storage, social networking sites, webmail, and online business applications. The cloud computing model allows access to information and computer resources from anywhere that a network connection is available. Cloud computing provides a shared pool of resources, including data storage space, networks, computer processing power, and specialized corporate and user applications.

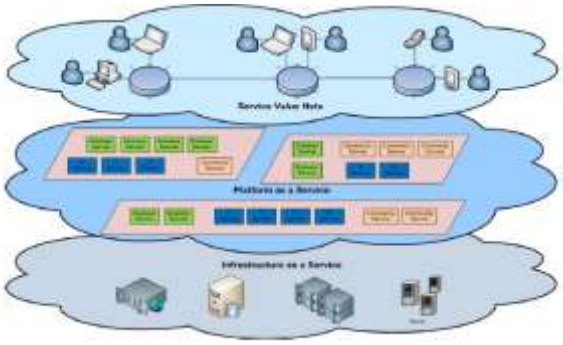

Figure 1.1 Cloud Architecture

Cloud Storage

Several large Web companies (such as Amazon and Google) are now exploiting the fact that they have data storage capacity that can be hired out to others. This approach, known as cloud storage allows data stored remotely to be temporarily cached on desktop computers, mobile phones or other Internet-linked devices. Amazon's Elastic Compute Cloud (EC2) and Simple Storage Solution (S3) are well known examples.

1.2 WIRELESS SENSOR NETWORKS (WSN)

Wireless Sensor Networks (WSN) has been a focus for research for several years. WSN enables novel and attractive solutions for information gathering across the spectrum of endeavour including transportation, business, health-care, industrial automation, and environmental monitoring. Despite these advances, the exponentially increasing data extracted from WSN is not getting adequate use due to the lack of expertise, time and money with which the data might be better explored and stored for future use. The next generation of WSN will benefit when sensor data is added to blogs, virtual communities, and social network applications. This transformation of data derived from sensor networks into a valuable resource for information hungry applications will benefit from techniques being developed for the emerging Cloud Computing technologies. Traditional High Performance Computing approaches may be replaced or find a place in data manipulation prior to the data being moved into the Cloud. In this paper, a novel infrastructure is proposed to integrate the Cloud Computing model with WSN.

The wireless sensor network (WSN) is becoming a very popular technology. Wireless which is comprised on a number of numerous sensor and they are interlinked or connected with each other for performing the same function collectively or cooperatively for the sake of checking and balancing the environmental factors. This type of networking is called as wireless sensor networking. A wireless sensor network(WSN) consists of a group of self organizing, lightweight sensor nodes that are used to cooperatively monitor temperature ,sound, humidity, vibration, pressure and motion. Each sensor node in a WSN is equipped with a radio transmitter, several sensor, a battery unit and a microcontroller.

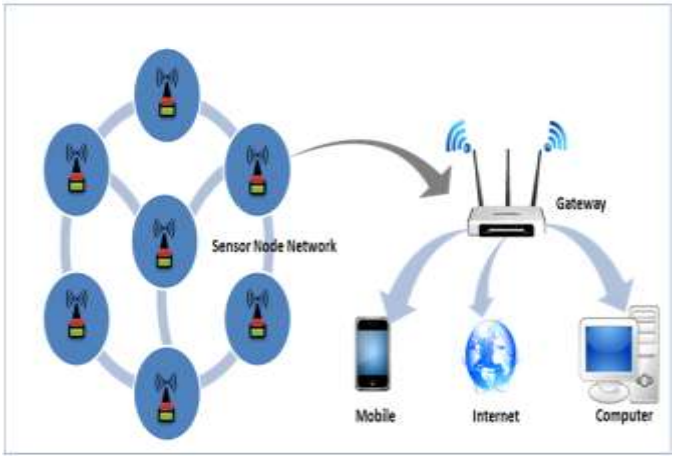

Figure 1.2 Wireless Sensor Network

WSNs are autonomous systems consisting of tiny sensors. These are associated with integrated sensing, limited battery lifetime, resource constraints and limited range.

Cloud computing is the delivery of computing services over the Internet. Cloud services allow individuals and businesses to use software and hardware that are managed by third parties at remote locations. This thesis proposes the comparison of wireless sensor network integrated cloud computing using middleware services

1.3 WIRELESS SENSOR NETWORK INTEGRATED WITH CLOUD

Connecting to wireless devices is not the easiest task to do. This problem is for small scale industries rather than larger business since larger companies have well structured network thus making wireless connections easy. Sometimes certain software are designed to relate to certain PCs alone in that case even usage of software maybe a problem. WSN have generated tremendous interest among researchers these years because of their potential usage in a wide variety of applications. Wireless sensor networks consist of a large number of small scale nodes capable of limited computation, wireless communication and sensing. WSN supports a wide range of applications like object tracking, infrastructure monitoring, habitat monitoring, battle field monitoring, health care monitoring etc. Sensor nodes consist of five main components:

- Computing Unit

- Communication Unit
- Sensing Unit
- Memory Unit
- Power supply Unit

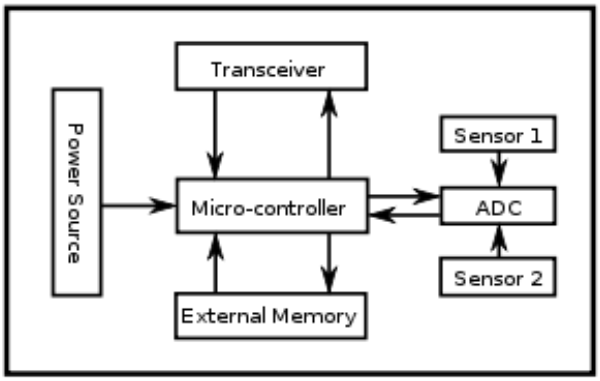

Figure 1.3 Sensor Node Structure

Developing applications for WSN is a transfer's job as the application developers have to meet considerable number of constraints due to the rigid integration of sensor nodes to the physical world. Designing a middleware is a novel approach for addressing these constraints wherein the middleware can act as binding software between applications and operating systems (OS).

The necessity of designing middleware software for WSN is to bridge the gap between the high level requirements from applications and the complexity of the operations in the underlying network; there are some other issues which could be well addressed by designing a middleware. The emergent dissemination of sensor networks and cloud-computing services has brought new opportunities of sensor-cloud integration that will facilitate users not only to monitor their objects of interest through sensors but also to analyze future status of these objects on the fly by using cloud services. For instance in agriculture, crop growers need to monitor real-time orchard temperature for frost protection purposes as well as to forecast the overnight temperature transition by applying sensor data to temperature-prediction models.

1.3.1 Architecture of Sensor - Cloud

Cloud computing service framework delivers the services of shared network through which the users are benefited by the services, and they are not concerned with the implementation details of the services provided to them. When a user requests, the service instances (e.g., virtual sensors) generated by cloud computing services are automatically provisioned to them.

There exists no application that can make use of every kind of physical sensors at all times; instead, each application required pertinent physical sensors for its fulfilment. To realize this concept, publish/subscription mechanism is being employed for choosing the appropriate physical sensor. In multiple sensor networks, every sensor network publishes its sensor data and metadata. The metadata comprises of the types, locations, and so forth for the physical sensors. Application either subscribes to one or maybe to more sensor networks to retrieve real-time data from the physical sensors by allowing each application to opt for the appropriate physical sensors' type.

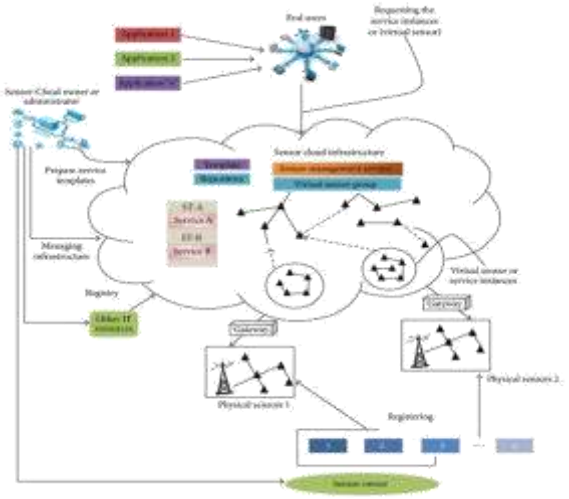

Figure 1.4 Sensor-Cloud System Architecture

Normally physical sensors focused on routing, clock synchronization, data processing, power management, OS, localization, and programming. However, few studies concentrate on physical sensor management because these physical sensors are bound closely to their specific application as well as to its tangible users directly. However, users, other than their relevant sensor services, cannot use these physical sensors directly when needed. Therefore, these physical sensors should be supervised by some special sensor-management schemes. The Sensor-Cloud infrastructure would subsidize the sensor system management, which ensures that the data-management usability of sensor resources would be fairly improved.

1.4 WSN MIDDLEWARE DESIGN

In heterogeneous WSN, sensor nodes have different characteristics, like different processing power, amount of memory and available energy. In order to meet the needs of their customers, and to develop their infrastructure to benefit from the technological advances in the field of WSN, most of independent administrations and companies deploy their own monitoring infrastructure and software architecture. In the case of a mismatch in data formats and structure exchange between nodes, the system should provide a mechanism for heterogeneous nodes to handle mismatch data, since all nodes communicate only with nodes of a similar data structure and exchange data formats model.

The mismatching of communication types exists due to the implantation of different formats of data. The following figure illustrates some WSN applications designed for healthcare, road traffic monitoring and environmental monitoring and as demonstrated, the sensed information coming from heterogeneous sensors is only reachable through specific application services via the company solution.

Figure 1.5. Architecture design of WSN applications.

Generally, the middleware performs the role of a translator that fills up the gap between the high level requirements of different applications running on wireless sensor networks and the complexity of different operations in the underlying sensor node hardware [4]. Figure 2 represents a logical architecture of a middleware within a WSN, on the one hand, on the sensor side, the middleware needs to deal with many challenges related to WSN characteristics and on the other hand, on the end user side, it should deal with the applications characteristics.

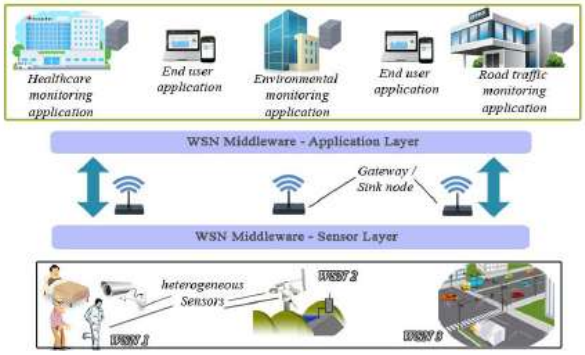

Figure 1.6 Logical Architecture of a Middleware

1.4.1 WSN- Cloud Computing Platform

The following figure consists of WSNs (i.e. WSN1, WSN2, and WSN3), cloud infrastructure and the clients. Clients seek services from the system. WSN consists of physical wireless sensor nodes to sense different applications like Transport Monitoring, Weather Forecasting, and Military Application etc. Each sensor node is programmed with the required application. Sensor node also consists of operating system components and network management components. On each sensor node, application program senses the application and sends back to gateway in the cloud directly through base station or in multi-hop through other nodes. Routing protocol plays a vital role in managing the network topology and to accommodate the network dynamics. Cloud provides on-demand service and storage resources to the clients. It provides access to these resources through internet and comes in handy when there is a sudden requirement of resources.

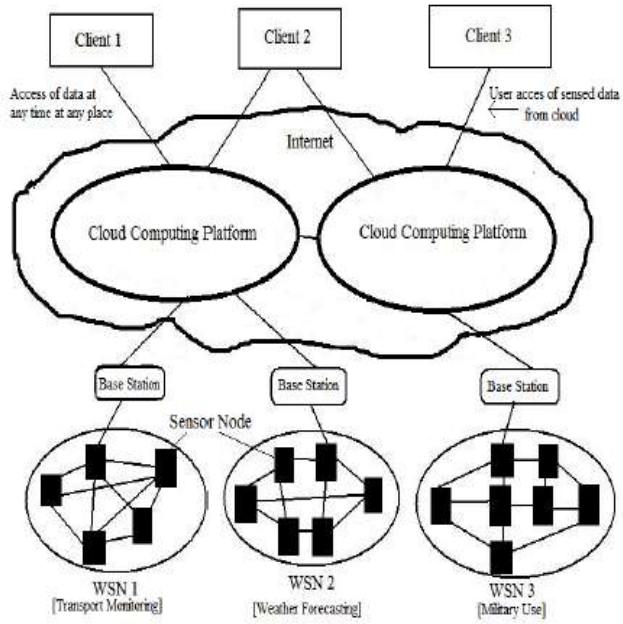

Figure 1.7 WSN- Cloud Computing Platform

1.5 NETWORKED CONTROL SYSTEM

Today, the number of devices with powerful sensing capabilities is consistently growing. Devices like smart phones are an amalgamation of multiple sensors. Mobile phones, sensors and electronic devices today capable of connecting to the internet and sharing information. However the sensors are heterogeneous in nature and there is a paucity of frameworks that are robust enough to handle the heterogeneity of the sensors. The sensors made by different vendors for different purposes differ not in functionalities but also in semantics, which makes integration heterogeneous sensors a challenge.

NCS can handle heterogeneous sensors and is capable of catering to different types of end users. For instance, smart cities of the future will have a distributed network of myriads of sensor nodes that measure different parameters for efficiently managing a city. The NCS will allow end users to view real time status of sensors they are in charge of. Each end user will be authenticated using their login credentials. The end users will be shown information strictly related to their domain. For example, when a traffic policeman logs in, he will have access to data from all the traffic related sensors in his circle.

1.6 NCARE SYSTEM

NCare has been developed specifically for healthcare professionals who manage patient's nutrition requirements in the community. NCare allows healthcare professionals the ability to register and then create patient profiles. These profiles are password protected via the secure NCare website governed by strict Nestlé Australia privacy policies. These profiles then allow the creation and management of a patient's nutrition regimen.

Once a patient's regimen has been created, the frequency of product delivery can be customised. The details are sent via secure data link or manual pdf to the appropriate state distributor for management of payment and delivery to the patient's doorstep. NCare allows healthcare professionals the ability to simply create nutritional regimens for their community patients and adjust them if required. NCare facilitates a seamless patient management process which allows healthcare professionals and their patients a clear and simple approach to nutritional supplementation in the community. NCare has specific reporting functionality which allows healthcare professionals the ability to create customized reports that can help with selected patient management, institution patient management and budget controls.

1.6 APPLICATIONS OF WSN

Earth/Environmental monitoring

It has evolved to cover many applications WSN to earth science research. This includes sensing volcanoes, oceans, glaciers, forests etc. some of the major areas are listed below.

- **Air Quality Monitoring**

The degree of population in the air has to be measured frequently in order to safeguard people and the environment from any kind of damages due to air pollution. In dangerous surroundings, real time monitoring of harmful gases is an important process because the weather can change rapidly changing key quality parameters.

- **Interior Monitoring**

Observing the gas levels at vulnerable areas needs the usage of high-end, sophisticated equipment, capable to satisfy industrial regulations. Wireless internal monitoring solutions facilitate keep tabs on large areas as well as ensure the gas concentration degree.

- **Exterior Monitoring**

External air quality monitoring needs the use of precise wireless sensors, rain and wind resistant solutions as well as energy reaping methods to assure extensive liberty to machine that likely to have tough access.

- **Air Pollution Monitoring**

Wireless sensor networks have been deployed in several cities to monitor the concentration of dangerous gases for citizens. These can take advantages of the ad hoc wireless links rather than wired installations, which also make them more mobile for testing readings in different areas.

- **Forest Fire Detection**

A network of sensor nodes can be installed in a forest to detect when a fire has started. The nodes can be equipped with sensors to measures temperature, humidity and gases which are produced by fire in the trees or vegetation.

- **Landslide Detection**

A landslide detection system makes use of a wireless sensor network to detect the slight movements of soil and changes in various parameters that may occur before or during a landslide. Through the data gathered it may be possible to know the occurrence of landslide long before it actually happens.

- **Water Quality Monitoring**

Water quality monitoring involves analyzing water properties in dams, rivers, lakes & oceans, as well as underground water reserves.

- **Natural Disaster Prevention**

Wireless sensor networks can effectively act to prevent the consequences of natural disasters, like floods. Wireless nodes have successfully been deployed in rivers where changes of the water levels have to be monitored in real time.

Industrial Monitoring

- **Machine Health Monitoring**

Wireless sensor network have been developed for machinery condition-based maintenance (CBM) as they offer significant cost savings and enable new functionality. In wired system, the installation of enough sensors is often limited by the cost of wiring. Previously inaccessible locations, rotating machinery, hazardous or restricted areas, and mobile assets can now be reached with wireless sensors.

- **Data Logging**

Wireless sensor networks are also used for the collection of data for monitoring of environmental information, this can be as simple as the monitoring of the temperature in a fridge to the level of water in overflow tanks in nuclear power plants. The statistical information can then be used to show how systems have been working. The advantage of WSNs over conventional loggers is the "live" data feed is possible.

- **Industrial Sense and Control Applications**

In recent research a vast number of wireless sensor network communication protocols have been developed. These new aspects are considered as an enabler for future applications in industrial and related wireless sense and control applications, and partially replacing or enhancing conventional wire-based network by WSN techniques.

- **Water /Waste Water Monitoring**

Monitoring the quality and level of water includes many activities such as checking the quality of underground or surface water and ensuring a country's water infrastructure for the benefit of both human and animal.

- **Agriculture**

Using wireless sensor networks within the agricultural industry are increasing common using a wireless network frees the farmer from the maintenance of wiring in a difficult environment. Gravity feed water systems can be monitored using pressure transmitters to monitor water tank levels, back to a central control center for billing. Irrigation automation enables more efficient water use and reduces waste.

- **Passive Localization and Tracking**

The application of WSN to the passive localization and tracking of non-cooperative targets (i.e., people not wearing any tag) has been proposed by exploiting the pervasive and low-cost nature of such technology and the properties of the wireless links which are established in a meshed WSN infrastructure.

- **Smart Home Monitoring**

Monitoring the activities performed in a smart home is achieved using wireless sensors embedded within everyday objects forming a WSN. A state change to objects based on human manipulation is captured by the wireless sensors network enabling activity-support services.

- **Area Monitoring**

The WSN is developed over a region where some phenomenon is to be monitored. A military example is the use of sensors detects enemy intrusion; a civilian example is the geo-fencing of gas or oil pipelines.

1.7 SECURITY IN CLOUD

To secure data, most systems use a combination of techniques:

• Encryption

A complex algorithm is used to encode information. To decode the encrypted files, a user needs the encryption key. While it's possible to crack encrypted information, it's very difficult and most hackers don't have access to the amount of computer processing power they would need to crack the code.

• Authentication processes

This requires a user to create a name and password.

• Authorization practices

The client lists the people who are authorized to access information stored on the cloud system. Many corporations have multiple levels of authorization. For example, a front-line employee might have limited access to data stored on the cloud and the head of the IT department might have complete and free access to everything**.**

Encryption and authentication are two security measures that can be used to keep your data safe on cloud storage provider

Figure 1.8 Securities in Cloud

Security Concerns

As with so many other technical choices, security is a two-sided coin in the world of cloud computing.

Privacy Concerns with a Third Party

The first and most obvious concern is for privacy considerations. That is, if another party is housing all our data, how do we know that it's safe and secure? we really don't. As a starting point, assume that anything we put on the cloud can be accessed by anyone. There are also concerns because law enforcement has been better able to get at data maintained on a cloud, more so than they are from an organization's servers.

Cloud providers should take care of the following persons for security concerns.

1. Hackers
2. Bot Attackers

Cloud Providers are taking more care of security concerns toward the data and its security for end users. The security Benefits are

- **Centralized Data**

There are some good security traits that come with centralizing your data.

- **Reduced Data Loss**

More than 12,000 laptops are lost in American airports every year. It's bad enough to lose your data, but it's especially bad for companies who lose proprietary data or other mission-critical information. Also, how many laptops employ really strong security measures, like whole-disk data encryption? If the laptop can be effectively compromised, the information will be in the hands of the thief.

By maintaining data on the cloud, employing strong access control, and limiting employee downloading to only what they need to perform a task, cloud computing can limit the amount of information that could potentially be lost.

- **Monitoring**

If your data is maintained on a cloud, it is easier to monitor security than have to worry about the security of numerous servers and clients

Logging In the cloud, logging is improved. Logging is usually thought of late in the game, and issues develop with storage space.

2. PROPOSED WORK

2.1 CLOUD

A Cloud system consists of 3 major components such as clients, datacenter, and distributed servers. Each element has a definite purpose and plays a specific role.

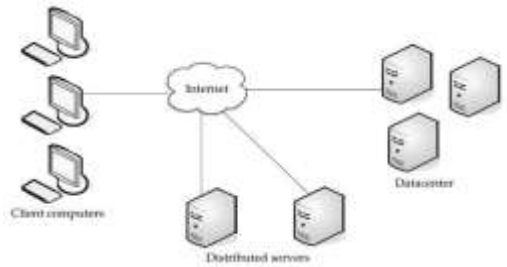

Figure 2.1 Three components make up a cloud computing solution

2.1.1 Clients

End users interact with the clients to manage information related to the cloud. Clients generally fall into three categories as given in:

- **Mobile:** Windows Mobile Smartphone, Smartphone, like a Blackberry, or an iPhone.
- **Thin:** They don't do any computation work. They only display the information.

  Servers do all the works for them. Thin clients don't have any internal memory.
- **Thick:** These use different browsers like IE or Mozilla Firefox or Google Chrome to connect to the Internet cloud.

Now-a-days thin clients are more popular as compared to other clients because of their low price, security, low consumption of power, less noise, easily replaceable and repairable etc.

2.1.2 Data centre

Data centre is a collection of servers hosting different applications. An end user connects to the data centre to subscribe different applications. A data centre may exist at a large distance from the clients. Now-a-days a concept called *virtualization* is used to install software that allows multiple instances of virtual server applications.

2.1.3 Distributed Servers

Distributed servers are the parts of a cloud which are present throughout the Internet hosting different applications. But while using the application from the cloud, the user will feel that he is using this application from its own machine.

2.1.4 Type of Clouds

Based on the domain or environment in which clouds are used, clouds can be divided into 3 categories:

- Public Clouds
- Private Clouds
- Hybrid Clouds (combination of both private and public clouds)

2.1.5 Virtualization

It is a very useful concept in context of cloud systems. Virtualization means "some-thing which isn't real", but gives all the facilities of a real. It is the software implementation of a computer which will execute different programs like a real

machine. Virtualization is related to cloud, because using virtualization an end user can use different services of a cloud. The remote data centre will provide different services in a full or partial virtualized manner.

The two types of virtualization found in clouds are :

- Full virtualization
- Para virtualization

Full Virtualization

In case of full virtualization a complete installation of one machine is done on the another machine. It will result in a virtual machine which will have all the software's that are present in the actual server.

Figure 2.2 Full Virtualization

Here the remote data canter delivers the services in a fully virtualized manner. Full virtualization has been successful for several purposes as pointed out in:

- Sharing a computer system among multiple users
- Isolating users from each other and from the control program
- Emulating hardware on another machine

Para Virtualization

In Para virtualization, the hardware allows multiple operating systems to run on single machine by efficient use of system resources such as memory and processor. e.g. VMware software. Here all the services are not fully available, rather the services are provided partially.

Figure 2.3 Para virtualization

Para virtualization has the following advantages:

- **Disaster recovery:** In the event of a system failure, guest instances are moved to hardware until the machine is repaired or replaced.
- **Migration:** As the hardware can be replaced easily, hence migrating or moving the different parts of a new machine is faster and easier.
- **Capacity management:** In a virtualized environment, it is easier and faster to add more hard drive capacity and processing power. As the system parts or hardware's can be moved or replaced or repaired easily, capacity management is simple and easier.

2.2 SERVICES PROVIDED BY CLOUD COMPUTING

The term services in cloud computing is the concept of being able to use reusable, fine grained components across a vendor's network. This is widely known as "as a service."

Offerings with as a service as a suffix include traits like the following:

- Low barriers to entry, making them available to small businesses
- Large scalability
- Multitenancy, which allows resources to be shared by many users
- Device independence, which allows users to access the systems on different hardware

NIST (National Institute of Standards and Technology) broadly divided cloud services into three categories or service models:

- Infrastructure-as-a-Service (IaaS): Includes the entire infrastructure stack i.e. servers, software, data centre space, virtualization platforms and network equipment.
- Platform-as-a-Service (PaaS): Sits on top of IaaS and adds an additional layer with application development capabilities, and programming languages and tools supporting the complete lifecycle of building and delivering applications and services over a cloud infrastructure.
- Software-as-a-Service (SaaS): Builds upon IaaS and PaaS and provides a self-contained operating environment delivering presentation, application and management capabilities.

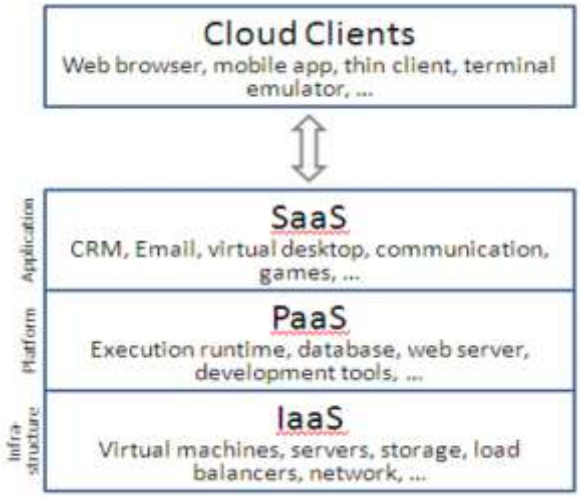

Figure 2.4 Cloud Computing Service Models

2.3 CLOUD COMPUTING AND CLOUD MIDDLEWARE

A cloud computing system can roughly be bifurcated into two sections: the front end and the back end. The front end is the client and back end includes the cloud services offered by the system. The front end includes the client's computer (or computer network) and the application required to access the cloud computing system. The back end of the system consists of various computers, servers and data storage systems that create the "cloud" of computing services. Theoretically, a cloud computing system could include practically any computer program you can imagine, from data processing to video games. Ideally, each application is assigned its own dedicated server. A central server administers the system, monitoring traffic and client demands to ensure everything runs smoothly. It follows a set of rules called protocols and uses a special kind of software called middleware. Middleware allows networked computers to communicate with each other. Usually servers do not run at full capacity. That means there's unused processing power going to waste. The technique is called server virtualization. A cloud computing system uses a technique called server virtualization to reduce the need for more physical machines. Virtualization is an illusion created by a server of having multiple servers, each having its own independent operating system.

Cloud Middleware

Cloud middleware is a software that enables seamless integration between various services, to harness the maximum potential of the available resources. It is the integration software that is hosted on a network to be readily available for interconnection of various software components or applications that are a part of the cloud. The cloud middleware is a major component of cloud computing as complex applications on a cloud need to work in unison. This software facilitates the re-use of applications and web components that are hosted anywhere on the cloud. Cloud middleware consists of two components, the user-level middleware and the core middleware. The user-level middleware provides tools and environments for cloud programming such as mashups, workflows, libraries and web 2.0 interfaces. Core middleware offers platforms such as Virtual Machines, VM management and deployment. The resource management layer is used to manage the resources. The

resource management layer also coordinates the resource sharing based on application needs, passed through the upper layers. Services provided by upper layers may need some resource sharing support, which is encapsulated in the communication layer. As an application uses such a service, the corresponding layer asks for the communication layer to manage the access control of the required resources. Indeed, the resource management layer commands the allocation and adaptation of resources, such that the QoS requirements specified by the applications can be met.

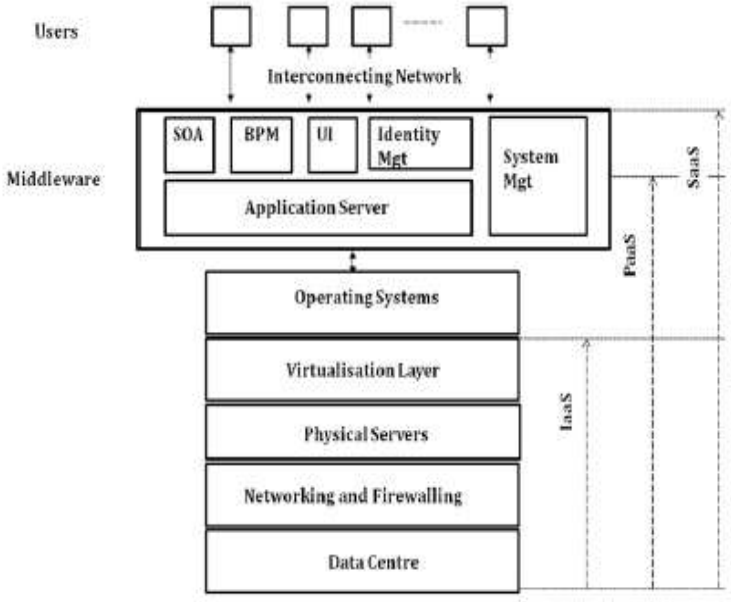

Figure 2.5 Cloud Middleware.

2.4 SENSOR-CLOUD INTEGRATION FRAMEWORK

WSN and Cloud Computing. The objective of the integration framework is to facilitate the shift of data from WSN to the Cloud Computing environment so that the scientifically and economically valuable data may be fully utilised. The framework components include: Data Processing Unit (DPU), Pub/Sub Broker, Request Subscriber (RS), Identity and Access Management Unit (IAMU), and Data Repository (DR). Data collected from the WSN moves through a gateway to the DPU. The DPU will process the data into a storage format and then send the data to the DR. Users will connect to the Cloud through the secured IAMU and will be given access on the basis of the policy stored against their user account. After access has been granted users can put forward data access requests.

The requests will be forwarded to the RS and the RS will create a subscription on the basis of this request and forward this subscription to the Pub/Sub Broker. Data received in the cloud will be identified by the DPU which will create a published data event and send the event to an event queue at the Pub/Sub Broker. When a new event is published, each subscription is evaluated by the event matcher. Once the event matching process finds a match the published data is made available to the user after further processing is carried out if required.

The framework components include: Data Processing Unit (DPU), Pub/Sub Broker, Request Subscriber (RS), Identity and Access Management Unit (IAMU), and Data Repository (DR). Data collected from the WSN moves through a gateway to the DPU. The DPU will process the data into a storage format and then send the data to the DR. Users will connect to the Cloud through the secured IAMU and will be given access on the basis of the policy stored against their user account. After access has been granted users can put forward data access requests.

The requests will be forwarded to the RS and the RS will create a subscription on the basis of this request and forward this subscription to the Pub/Sub Broker. Data received in the cloud will be identified by the DPU which will create a published data event and send the event to an event queue at the Pub/Sub Broker. When a new event is published, each subscription is evaluated by the event matcher. Once the event matching process finds a match the published data is made available to the user after further processing is carried out if required.

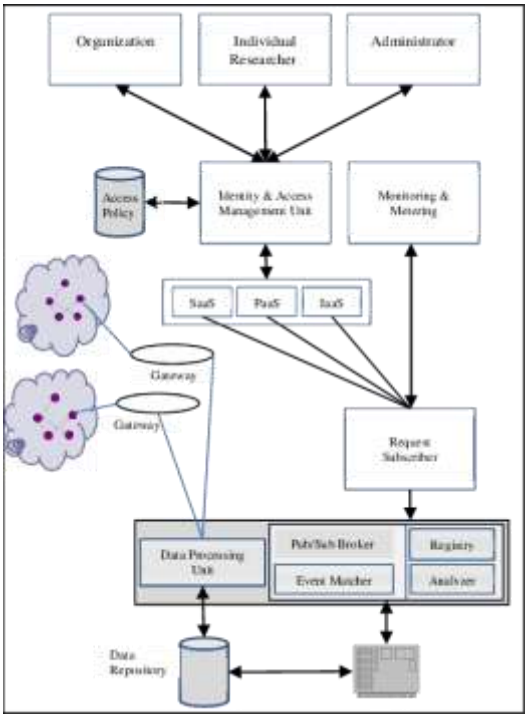

Figure 2.6 Sensor-Cloud Integration Framework

2.5 NETWORKED CONTROL SYSTEM (NCS) FRAMEWORK

The application of battlefield surveillance in WSN is to classify and detect the multiple targets to identify different categories like civilian, soldiers, enemies, wild animals, domestic animals. In such circumstances military operation is essential to identify the different categories and target to the enemies without upsetting the others. In this situation virtualization of sensor network plays an important role to sense the environment to identify civilian, soldiers, enemies, and animals. As a result, it only target to the enemies without affecting the others. To achieve this, WSN provide services to sense different environmental parameters such as sound, vibration, civilian *etc*., in a single sensor network deployment. Soldiers can supervise the battlefield to target the enemies by sensing sound, animal movement, and civilians by using different services of the WSNs.

Vehicle monitoring is one of the challenging issues in WSNs to track the vehicles, and monitor the traffic signals. Tracking of vehicle is one of the challenging issues now-a-days to control the crimes. The main objective of vehicle monitoring is to keep the status of the vehicle like current location, speed, light status, pollution control, weight control, status of driver, fuel status, distance covered, and time of arrival etc. by using services of the WSN. As a result, traffic police can track the target vehicle to avoid crimes in the smart cities. It can be achieved by incorporating the sensors in the vehicle. It can also monitor the traffic control system by considering density of individuals and sound to avoid the traffic.

Health care monitoring can be achieved by using wearable sensors such as temperature sensors, and accelerometer sensors etc. These sensors are required to collect patient's health related data such as tracking body temperature, heartbeat count, pulse rate, and blood sugar control etc. A set of sensors are deployed in the home or hospital to monitor the movement of doctors, nurse and other people. It can also keep track of patient's details like body temperature, heartbeat count, pulse rate, and blood sugar control *etc.*, and also alert an alarm in case of any deflections in patient's health parameters.

Power grid is a complex network and is required to monitor and control the power system efficiently. Smart grid is one of the emerging technology which helps in taking smart decision for automatic monitoring and control of power grid. It can be achieved by deploying the sensors throughout the grid to keep track of different components such as generation unit, transmission unit and distribution unit. WSNs enable different services on the sensor infrastructure to keep track smart decision to monitor and control of power system.

WSNs provide services for monitoring of industry. These services are responsible for different applications such as production, service, safety, and operation. In production unit different sensors are deployed to make the system autonomous and increase the production quantity and quality. Operation unit is also equipped with sensors to monitor the devices. If any hazards such as increase in speed or sound in any device occur, then it makes the workers alert by giving alarms.

Here we give a brief comparative of WSN integrated cloud in N care systems mainly focused on important metric Cloud Database, different Sensor data sharing platform and Cloud based sensor data processing.

Sensor-Clouds can be used for health monitoring by using a number of easily available and most often wearable sensors like accelerometer sensors, proximity, ambient light and temperature sensors, and so forth to collect patient's health-related data for tracking sleep activity pattern, blood sugar, body temperature, and other respiratory conditions. These wearable sensor devices must have support of BWI (Bluetooth's wireless interface), UWB (Ultra wideband), and so forth interface for streaming of data and are connected wirelessly to any smartphone through this interface.

These smart phone devices pretend to function like a gateway between the remote server and sensor through the Internet, maybe GPRS/Wi-Fi, or other sort of gateways. To transform this system into services-based structure, web-services-based interfaces are used by smart phone device to connect to the server. The system prototype should have made to be robust, mobile, and scalable. Robust in the sense means that it should recover itself from circumstances, which may lack connectivity issues due to power (i.e., battery), failure, or gateway cutoff to patient's wearable devices. Mobile in the sense means that it should be capable of tracking signals into heterogeneous environments; that is, it must catch the signals irrespective of whether the patient went outside or still resided into the hospital/building. It should be scalable so that it could be deployed easily for several users concurrently without affecting the performance metrics.

Finally, such prototype system should be retargeting able and extensible in nature. Re-targetable refers to the fact that it can handle various displays with distinct form factors and screen resolution. It means that the same health applications can be displayed to any smartphone display like PDA (personal digital assistant) or to a bigger console device in a hospital where doctors, helpers, or nurses may track the acquired data or processed information from distance. The extensibility aspect requires that if any newer sensing devices are introduced into the system for acquiring the patient's health-based information, the system should function efficiently and conveniently without affecting backend server of the services. In this platform, context awareness can be achieved that can direct us to derive a better level of emergency services to the patient. The information regarding recent operational laboratories, missing doses of pills, number of handicaps, and other situations would be helpful in health monitoring.

The system should not adhere to any changes made into the operating system or intermediate components of sensing devices and is designed in such a way that it would cause minimal disturbance to services provided to existing end users of the system.

2.5.1 Features

NCS has the following features

- User Level NCS on stock application
- Cloud setup and application deployment
- Getting cloud statistics and performance evaluation of each node
- Resource Monitoring of cloud nodes
- Deploying an application war file on cloud nodes considering their CPU, RAM usage using cloud controller

3. PROPOSED ALGORITHM

3.1 HONEY BEE FORAGING ALGORITHM

This algorithm is derived from the behavior of honey bees for finding and reaping food. There is a class of bees called the forager bees which forage for food sources, upon finding one, they come back to the beehive to advertise this using a dance called waggle dance. The display of this dance, gives the idea of the quality or quantity of food and also its distance from the beehive. Scout bees then follow the foragers to the location of food and then began to reap it. They then return to the beehive and do a waggle dance, which gives an idea of how much food is left and hence results in more exploitation or abandonment of the food source.

In case of NCare System, as the web servers demand increases or decreases, the services are assigned dynamically to regulate the changing demands of the user. The servers are grouped under virtual servers (VS), each VS having its own virtual service queues. Each server processing a request from its queue calculates a profit or reward, which is analogous to the quality that the bees show in their waggle dance. One measure of this reward can be the amount of time that the CPU spends on the processing of a request. The dance floor in case of honey bees is analogous to an advert board here.

Each of the servers takes the role of either a forager or a scout. The server after processing a request can post their profit on the advert boards with a probability of pr. A server can choose a queue of a VS by a probability of px showing forage/explore behavior, or it can check for advertisements (see dance) and serve it, thus showing scout behavior. A server

serving a request, calculates its profit and compare it with the colony profit and then sets its px. If this profit was high, then the server stays at the current virtual server; posting an advertisement for it by probability pr. If it was low, then the server returns to the forage or scout behavior.

Algorithm

1. for i=1,…,ns
   scout[i]=Initialise_scout()
   flower_patch[i]=Initialise_flower_patch
   (scout[i])

2. do until stopping_condition=TRUE
   Recruitment()
   for i =1,…,nb

   flower_patch[i]=

   Local_search(flower_patch[i])

   flower_patch[i]=

   Site_abandonment(flower_patch[i])

3. flower_patch[i]=Neighbourhood_shrinking

   (flower_patch[i])

   for i = nb,…,ns
   flower_patch[i]=
   Global_search(flower_patch[i])}

The algorithm starts with the n scout bees being placed randomly in the search space. The fitness of the sites visited by the scout bees are evaluated in step 2.

1. Initialize population with random solutions.
2. Evaluate fitness of the population.
3. While (stopping criterion not met) //Forming new population.
4. Select sites for neighborhood search.
5. Recruit bees for selected sites (more bees for best e sites) and evaluate fitnesses.
6. Select the fittest bee from each patch.
7. Assign remaining bees to search randomly and evaluate their fitnesses.
8. End While.

In step 4, bees that have the highest fitnesses are chosen as “selected bees” and sites visited by them are chosen for neighborhood search. Then, in steps 5 and 6, the algorithm conducts searches in the neighborhood of the selected sites, assigning more bees to search near to the best e sites. Searches in the neighborhood of the best e sites which represent more promising solutions are made more detailed by recruiting more bees to follow them than the other selected bees. Together with scouting, this differential recruitment is a key operation of the Bees Algorithm.

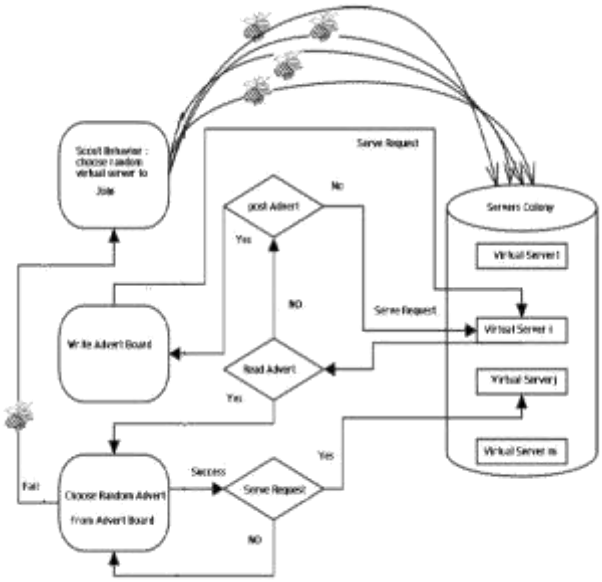

Figure 3.1 Server Allocations by Foraging in Honey bee technique

3.1 The Load Balancing algorithm with Honey Bees Strategy

A load balancing algorithm based on honey bee behavior (LBA_HB) is proposed. Its main goal is distribute workload of multiple network links in the way that avoid underutilization and over utilization of the resources. This can be achieved by allocating the incoming task to a virtual machine (VM) which meets two conditions; number of tasks currently processing by this VM is less than number of tasks currently processing by other VMs and the deviation of this VM processing time from average processing time of all VMs is less than a threshold value.

Load balancing is the process of distributing workloads and computing resources in a cloud computing environment. It allows enterprises to manage application or workload demands by allocating resources among multiple computers, networks or servers. Load balancing is often used to avoid the bottleneck, so that several characteristics of load balancing can be achieved such as: equal division of tasks across all hosts, facilitation in achieving service quality, improve overall performance of the system, reduce response time, and improve resource utilization

The following figure shows the load balancer of virtual machines (VMs). It assigns multiple tasks to VMs that execute them simultaneously by a way that guarantees a balance between these VMs. The primary goal of load balancing in a cloud environment is to balance the workload of the hosts in proportion to their capacities, which is measured in terms of their processor speed, available memory space, and bandwidth.

Figure 3.2 Virtual Machine Load Balancing

Load balancing algorithms are classified into two types; static and dynamic. Static algorithms are much simpler as compared to dynamic algorithms. Static algorithms work properly only when hosts have low variations in the load, since they do not take into account the previous state or the behavior of a host while distributing the load. Dynamic load balancing algorithms are more suitable for widely distributed systems such as cloud computing. Round robin (RR) is a well-known straightforward static scheduling algorithm. It allocates tasks to each node in turn, without considering the resource quantity of each VM and the execution time of tasks. Modified throttled algorithm is a dynamic load balancing algorithm that uniformly distributes the incoming tasks among available VMs. However it doesn't consider resource utilization during task allocation.

In this paper, a Load Balancing Algorithm based on Honey Bee behavior (LBA_HB) is proposed. It is completely inspired by the natural foraging behavior of honey bees. The allocated task updates the remaining tasks about the VM status in a manner similar to the bees finding an abundant food source, updating the other bees in the bee hive through its waggle dance. The proposed LBA_HB algorithm has been simulated using CloudSim . The proposed algorithm is compared with both conventional and SI based load balancing algorithms; round robin, modified throttled, ant colony, and honey bee algorithms. The results of experiments show the efficiency of LBA_HB in terms of response time, makespan, standard deviation of load, and degree of imbalance.

The Pseudo code shows the main processes of the LBA_HB

Input: List_statehost , List_stateVM Output: VM(j) // return the number (i) of host which have minimum processing time // i is the number of specified host

1: i ← -1

2: minPT← Integer.MAX_VALUE

3: For each host(i) in List_statehost

4: If host(i) available then

5: If (PThost(i) < minPT) then

6: minPT = PThost(i)

7: i ← number of the current host

8: End if

9: End if

10: End for // return the number (j) of VM which has minimum count of requests // j is the number of specified VM

11: j ← -1

12: mincount← Integer.MAX_VALUE

13: For each VM(j) in List_VMhost(i)

14: If VM available then

15: If (Count_REQVM(j)< mincount) then

16: mincount = Count_REQVM(j)

17: j ← number of the current VM

18: End if

19: End if

20: End for

21: if (j=-1) then

22: append coming task in waiting queue until one VM become available.

23: else

24: Allocate the task to VM(i).

25: Update allocated information i.e.; how many tasks are being processed, current processing time of the host and VM and check the availability of VM and host

26: De-allocate the task from this VM after end of the task execution.

27: Update allocated information i.e.; how many tasks are being processed, current processing time of the host and VM and check the availability of VM and host.

28: end if

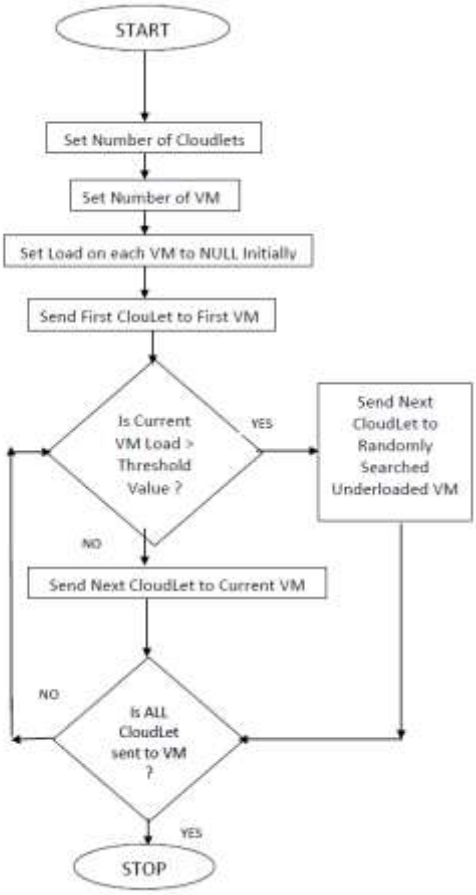

Figure 3.4 Flow Chart of Load Balancing in Cloud Computing based on Honey Bee Behaviour

The technique is proposed for cloud computing environments predicated on deportment of honey bee foraging strategy. This algorithm not only balances the load, but takes into consideration the priorities of tasks that have been abstracted from heavily loaded Virtual Machines. The tasks abstracted from these VMs are treated as honey bees, which are the information updates globally.

This algorithm withal considers the priorities of the tasks. Honey bee deportment inspired by NCS ends the overall throughput of processing and priority predicated balancing fixates on reducing the duration a task has to wait on a queue of the VM. Thus, it reduces the replication of time of VMs. We have compared our proposed algorithm with other existing techniques.

The NCS technique works well for heterogeneous cloud computing systems and is for balancing non-preemptive independent tasks. We can elongate this kind of NCS for workflows with dependent tasks. This algorithm considers priority as the main QoS parameter. In future, they orchestrate to ameliorate this algorithm by considering other QoS factors also.

4. CONCLUSIONS AND FUTURE WORK

4.1 CONCLUSION

Integration of WSN and Cloud Computing will provide benefits to organizations and the research community. Organizations will benefit by utilizing Cloud storage and an optimized framework for processing, storage and retrieval of WSN generation data. The proposed WSN Cloud Computing framework will provide an optimal approach to user management, access control, storage and retrieval of distributed data.

WSNs hold the promise of many applications in the area of monitoring and control systems. Many properties of the environment can be observed by the monitoring system with the advent of cheap and tiny sensors. All these applications are meant for the specific purposes, and therefore maintaining data transport reliability is one of the major concern and the most important challenge. To address the reliability, to survey the various existing techniques; each of them has its own unique working to ensure the reliability. Some of the techniques use retransmission mechanism while others use redundant information for insuring the reliability. Few of the above objectives may be considered in the future by the researchers.

The NCS offers many advantages. The use of cloud infrastructure increases the computational power of the system. In such a system, computation is done using the cloud infrastructure rather than by individual sensor nodes. As a result the power requirements and size of each sensor can be reduced. Smaller sensors are easier to sustain in times of an emergency such as a natural calamity and to conceal for detecting crime. Additionally, NCS offers a high degree of scalability. As a result it can handle increase in number of sensor nodes without much performance overheads. Since the system is dynamic, back-up sensors can be enabled, in case the main sensors fail. NCS can be utilized to collect data from different types of heterogeneous sensors and to provide domain specific sensor data to the end users.

A load balancing algorithm in cloud computing environment based on behavior of honey bee foraging strategy is proposed. The proposed LBA_HB aims to minimize overall response time and data center processing time since it distributes workload between different VMs with considering availability and load of each VM. It limits allocation of requests to VM when the variation of this VM processing time from average processing time of all VMs becomes more than or equal to a predefined threshold.

4.2 FUTURE WORK

Future work will include further development of the data processing, storage and retrieval methodology. There are parallels to the on-demand video Cloud solutions currently being implemented. Another aspect of future research will be to identify an optimal approach to permit data manipulation prior to publishing.

Compared to the current state-of-the-art in building applications on Sensor Node, our approach enables an efficient deployment of several types of applications on Sensor Node thus allowing these resource-constrained platforms to achieve better performance with a controlled overhead. Our optimization has focused so far only on the client side and has assumed the server's resources to be infinite..

Sensors will play a key role in our future and it will become more and more important that we use the framework to integrate sensor data. The one thing that we could do to make this framework better is to use a probability model to find out the columns required to be sent to the server without them being manually specified in a descriptor. This concept would involve a lot of automation and make this framework easy to use.

Another improvement that can be done is to provide some sort of control over these cluster daemons to the administrators. This control would help in maintain a large sensor networks where handling the sensors becomes a real issue. A number of side parameters can also be sent about the health of these sensors to their geographical location which could prove useful.

REFERENCES

[1] R. Bloor. What is a cloud database. Technical report.

[2] S. Bose and R. Liu. Cloud computing complements wireless sensor networks to connect the physical world. Technical report.

[3] Chandrakant N, Bijil A P, Deepa Shenoy P, Venugopal K R, and L M Patnaik. Middleware service oriented rescue and crime information system (rcis) using heterogeneous fixed nodes in wsns. In ADCONS 2011, December 16-18, 2011, Karnataka, India.

[4] Chandrakant N, Bijil A P, Deepa Shenoy P, Venugopal K R, and L M Patnaik. Middleware service oriented rescue and crime information on cloud (rcic) using heterogeneous nodes in wsns. In ADCONS 2011, December 16-18, 2011, Karnataka, India, pages 1–5, 2012.

[5] Giurgiu, O. Riva, D. Juric, I. Krivulev, and G. Alonso. Calling the cloud: Enabling mobile phones as interfaces to cloud applications. In Proceedings of the 10th

[6] International Middleware Conference Middleware'09), November 30 December 4, 2009.

[7] D. Huang, X. Zhang, M. Kang, and J. Luo. Mobicloud: Building secure cloud framework for mobile computing and communication. In Service Oriented System Engineering (SOSE), 2010 Fifth IEEE International Symposium, pages 27 – 34, June 2010.

[8] Hung-Chin Jang, Yao-Nan Lien, and Tzu-Chieh Tsai. Rescue information system for earthquake disasters based on manet emergency communication platform. In IWCMC09, June 21 24, 2009, Leipzig, Germany.

[9] G. Kaefar. Cloud computing architecture.

[10] A. Khan and K. Ahirwar. Mobile cloud computing as a future of mobile multimedia database. In International Journal of Computer Science and Communication.

[11] D. Kovachev, Y. Cao, and R. Klamma. Mobile cloud computing: A comparison of application models. In Service Oriented

[12] System Engineering (SOSE), 2010 Fifth IEEE International Symposium. Madoka Yuriyama, Takayuki Kushida, (2010), "Sensor Cloud Infrastructure: Physical Sensor Management with Virtualized Sensors on Cloud Computing" 13th International Conference on Network-Based Information Systems, IEEE, 2010.

[13] C.O. Rolim, F.L. Koch, C.B. Westphall, J.Werner, A. Fracalossi, G.S. Salvador, (2010)," A Cloud Computing Solution for Patient's Data Collection in Health Care Institutions", 2nd Intl Conference on eHealth, Telelmedicine, and social medicine, 2010, pp. 95-99.

[14] Margaret O'Brien, (2008), "Remote Telemonitoring - A Preliminary Review of Current Evidence", European Center for Connected Health, 30thJune 2008 .

[15] BiswasJit, Jayachandran Maniyeri, Kavitha Gopalakrishnan, Shue Louis, PhuaJiliang Eugene, HenryNovianusPalit, Foo Yong Siang, Lau LikSeng, and Li Xiaorong, (2010), "Processing of wearable sensor data on the cloud – a step towards scaling of continuous monitoring of health and well-being" 32 Annual Intl Conference, IEEE EMBS, pp. 3860-3863, 2010.

[16] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss and P. Levis, (2009), "Collection Tree Protocol," The 7th ACM Conference on Embedded Networked Sensor Systems (SenSys 2009), 2009.

[17] Yasser Mesmoudi, Yasser El Khamlichi, Abderrahim Tahiri, Abdellah Touhafi, An Braeken, (2017), "A Middleware Based Service Oriented Approach for Wireless Sensor Network "International Journal of Advanced Computational Engineering and Networking", Volume-5, Issue-11, Nov.-2017

[18] Sanjit Kumar Dash, Subasish Mohapatra and Prasant Kumar Pattnaik, " A Survey on Applications of Wireless Sensor Network Using Cloud Computing", (2010), International Journal of Computer Science & Emerging Technologies, Volume 1, Issue 4, December 2010