Integrating Cloud-WSN To Analyze Weather Data And Notify SaaS User Alerts During Weather Disasters

Arjun D S
Department of Computer Science
4th Semester M.Tech, C-DAC Intern, RNSIT College
Bangalore, India
arjunds288@gmail.com

Arunachalam Bala
Senior Technical officer, SSDH
C-DAC
Bangalore, India
barunachalam@cdac.in

Dwarakanath V
Project Engineer, IOT Department
C-DAC
Bangalore, India
DWARAKANATHV@cdac.in

Sampada K S
Professor, Department of Computer Science
RNSIT College
Bangalore, India
k.s.sampada@gmail.com

Prahlada Rao B B
Joint Director, SSDH
C-DAC
Bangalore, India
prahladab@cdac.in

Haribabu Pasupuleti
Principal technical officer
C-DAC
Bangalore, India
hari@cdac.in

***Abstract*— This paper presents an enhanced architecture for integrating cloud with wireless sensor networks to analyze weather data and notify SaaS users alert during weather disasters at low cost. The occurrence of natural disasters affects lives, damages property and changes our lives completely. Existing system does not support node and network level virtualization for weather sensors. The proposed system overcomes the above limitation by deployment of WSN infrastructure for multiple weather applications using virtual sensor and overlay concept. Monitoring weather data and providing SaaS and social network disaster alerts based on decision ID3 technique and provide cloud authentication using secure shell. These factors improve and provide high quality disaster alters to users and weather analysts at low cost.**

Keywords*— *Wireless sensor network, Cloud computing, Weather disasters, Smart cities, SaaS

I. Introduction

Wireless sensor networks (WSN) form a communication network by interconnection of large distributed sensor nodes. The nodes are connected in adhoc fashion, operating independently of other sensor nodes. WSN can achieve low cost, small size, low power consumption, fewer network components and other features easily. In recent years it has been implemented in weather, agriculture, industry, military and other fields. C-DAC Wireless sensor network development kit (WSNDK) consists of Ubi-Sense designed with smoke, humidity, temperature, light sensors along with a buzzer and an Ubimote interface used for linking sensor boards supports Z-stack and Tiny OS.

Sensors have hardware limitation factors so people were in pursuit of a better performance and enhancing greater computing capability, people turn to find other techniques to overcome the limitation of the sensor hardware. The concept of "Cloud" came into existence. "Cloud" refers to a network. NIST version of cloud computing is classified as "cloud computing is a blueprint for enabling convenient on demand network usage of shared pool of configurable computing resources" [1]. Cloud computing has three architecture layers which are namely SaaS (Software as a service), IaaS (Infrastructure as a service) and PaaS (Platform as a service). A wide variety of board market solutions are provided by SaaS vendors through software products and hardware portal interface [2]. Cloud services can be employed in four deployment models such as community cloud, private cloud, hybrid cloud and public cloud [3]. C-DAC SuMegha Scientific cloud provides Storage-As-a-service through Cloud Vault, SaaS through Job Submission Portal, Map reduce and MPI clusters on demand and also solve complex business and engineering problems through high performance computing (HPC).

Global warming is one of the main reasons for increasing natural weather disasters. In the view of interest of the people we need to monitor weather sensor data and develop a disaster alert system. In some situations like earthquake, earth fall and flooding of water it is possible to caution within a specified period by raising alert. In the case of rainfall, snowfall alerts

978-1-4799-8047-5/15/$31.00

can be send to weather analysts for further analysis and prediction of climate. The existing surveillance system is a tool used to measure weather activities but it's difficult to install needs power supply and unable to connect to signal wiring. To overcome this issue a wireless sensor network was employed to help resolve data communication. But WSN alone does not support virtualization to support sharing of infrastructure by multiple weather applications. Here, we employ WSN node and network level virtualization to overcome the drawback.

In this paper we are proposing SaaS emergency management by cloud computing which includes data storage, data backups and alert application [3]. Emergency alerts can be sent through SMS, MMS, Email or social network. The purpose for using cloud computing is only computers and databases used to connect the internet are at risk this leads to lower recovery cost. The user data can be protected from disaster because cloud servers are far from disaster site placed inside a data center. Hence the recovery costs are considerably lower. In this paper we propose and use node and network level virtualization of weather sensors, local databases, Secure Shell (SSH) to provide authentication mechanism, ID3 technique to signal weather disaster alerts by matching with pre-conditions and providing SaaS alerts to users, weather analysts and social network.

II. Literature Survey

The weather surveillance equipment [4] is used to measure parameters like air pressure, vibration, moisture, sample temperature etc. These parameters fluctuate depending on the requirement of weather surveillance system. Monitoring system which is the tool used for weather detection is the core part of the work. Draw backs of this particular weather monitoring system devices are installed in different location, in some situations it is not easy to install weather equipment in some areas reasons being lack of access to power source and unable to connect to signal wiring and tools used for measuring are very costly.

Lim et al [5] proposed integrating a sensor network with grid computing (GRID) called sensor grids to store weather data. They are capable of enhancing the potential of these technologies with the help of (SPRING) Scalable proxy-based architecture for sensor grid but in the current application scenarios GRID may not be expropriate for most occasions. Since GRID intensify on elevated computing performance whereas, cloud computing targets on generic applications.

Wen Yaw [6] proposed cloud setup established on wireless sensor network uses big data to store data into distributed SQL databases that monitors real-time weather information but this approach does not provide SaaS disaster alerts to users and social network for emergency management.

J.A. Nazabal [7] proposed possession of KNX devices by employing USB/KNX interfaces for cumulating sensor data and remote tracking. This system has been developed in such a way that sensed data from KNX sensor devices uses IP packets to send data to MySQL server. Limitation is that sensory data is communicated using unencrypted text without any encryption mechanism. This is a threat because sensory data transit over the internet and could be accessible to undesirable recipients.

Round robin databases RRD [8] tool version 1.0 and MRTG are stored in circular buffers. The size and step of the buffer can be defined by users. Often a new sensed value is inserted into the circular buffer. When the circular buffer is completely full old sensor data gets overwritten with new sensor data. RRD uses small circular buffer for aggregating daily sensed data and larger circular buffer for weekly sensed data. Existing weather system precious sensor data that could have been useful for analysis has not been retained. Since, it's either replaced or lost.

TinyDB [9] is a query system for retrieving information from a network of TinyOS weather sensors and storing weather data. TinyDB accumulates data from the surrounding environment, aggregates, filter and routes it to a computer. TinyDB uses Java API for writing computer applications that extract data and query from the weather sensor network. Features of TinyDB include Complex Queries, Managing Metadata, multiple queries and Iterative deployment using Shared queries. Limitation of TinyDB is every node must wake up at every time step, data loss, no quality guarantee and inefficient data.

III. Proposed Architecture

In this paper keeping Xuan et.al SC3 model [10] as reference we propose SaaS integrated CLOUD-WSN framework to notify users weather disaster alert in which the system applies both node and network level virtualization to achieve maximum utility and sharing of WSNs. We consider three modules for this framework.

A. *WSN Node and Network Virtualization Module*

Node and Network level virtualization [11] are explained in detail in WSN Virtualization applied for weather applications in section IV. Weather Activities can be identified by cameras collecting video records and extracting the background data to get location instability based on weather parameters and comparing with predefined weather conditions for that particular area. Initially, weather data is captured from cameras and sensors, later data is transferred to the cloud gateway. Gateway assorts data into sensory weather data, complex weather event and video data and stores into a local database.

Noisy data and redundant sensory data are filtered during filtering process to decline transmission overhead before moving to Cloud and the local database data is rejuvenated with new filtered data. If end user sends a query requested through SaaS application, the response handler retrieves data from the local weather database and conveys to the concerned person. The proposed SaaS integrated CLOUD-WSN framework for weather data is shown in fig.1.

B. *SaaS Module*

Weather Logger offers low energy consumption and low cost weather sensor device. This device gathers many sorts of weather behavior patterns and accumulates it. Applying a quick and scalable weather sensor data dissemination mechanism the sensory weather data is uploaded into the Cloud. Inside the Cloud, this sensory data is weather, video data or detected complex weather activities such as rain, snowfall, earthquake, fire and so on. SaaS users much first authenticate themself to servers.

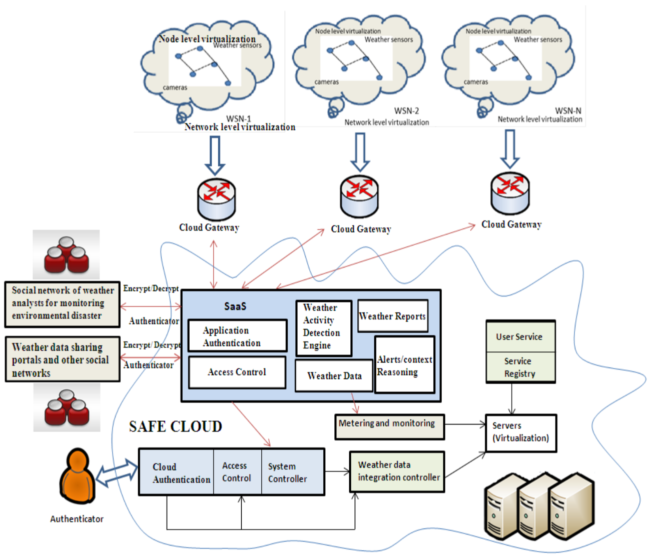

Fig.1. SaaS Integrated CLOUD-WSN Framework for Weather Data Alerts

Access control mechanisms are used to allow authorized users to access the servers. Activities are identified by gathering video records from camcorder and extricate the location background to identify instability based on weather parameters and comparing to predefined weather conditions for that particular area. Data is moved onto the Cloud. So, weather activity detection unit in the Cloud deduces weather activities. Context and weather activities are sent to the weather unit designed to infer complex activities and take decisions depending on location and send disaster alerts to SaaS users to move to safer locations. It also needs to take verdict to acknowledge distinct situations. Example, if earth quake or fire break disaster occurs alerts should be sent to SaaS users, weather analysts, social network and weather sharing portals. If rainfall or snowfall occurs weather analysts should be sent alerts so that they can analyze immediately the weather data and can predict future disasters such as cyclone or winter storms. If the weather analyst predicts danger SaaS alerts need to be sent to users, social networks and weather sharing portals. A SaaS user must request subscription [12] before weather disaster alerts are received via SMS, Email etc. The previous weather data analysis reports are stored for authorized SaaS users and analysts to access.

C. Cloud Module

Authorized users, weather analysts should authenticate themselves with C-DAC Scientific cloud servers [13] [14] [15] before they can access weather data. Authentication is done by using secure shell (SSH). C-DAC scientific cloud supports SSH authentication and x-server access control. On successful authentication the access control such as x-server used for registering users for different type of service and improve security and transparency of user's access and also to take decision whether to allow a user or not. If allowed the user can view the cloud weather data. Weather data is sent to authenticated users. The system controller which controls the cloud and system components and the weather data integration controller integrates data from different sources such as different weather data sharing sites and sends this data to weather analysts for analysis. Servers are virtualized for effective usage and sharing of resources.

IV. WSN VIRTUALIZATION APPLIED FOR WEATHER APPLICATION

To avoid redundant deployment of weather sensor we apply Node level virtualization and Network level virtualization [16]. Node level virtualization can be attained by concurrent execution of weather tasks from multiple weather applications by the sensor nodes. Network level virtualization is the ability of WSN weather sensor nodes to dynamically form a group to perform isolated and transparent execution of weather application tasks in a way that each set belongs to a different weather application. Author Fatna et.al [16] four principle idea to apply WSN virtualization is imposed to our proposed framework.

- First principle, new weather application (e.g. weather application) is installed as a new weather overlay on top of physical Wireless Sensor Network. Advantages of weather overlay are: lack central control, distributed and allow sharing of resources [17].
- Second principle, any physical weather sensor node can implement a task for a given weather application installed in the weather overlay. Weather tasks involve collecting weather sensor data and sending weather event notifications to weather overlay applications.
- Third principle is that the overlay-connected operations are not certainly performed by the weather sensor nodes directly concerned, as enough capabilities may not be there to support the weather overlay middleware. If that is the case, they will transfer the action to more dynamic weather sensors and other nodes.
- Last principle guideline is that within the architecture design there are distinct paths: signaling and data. The weather sensor data like temperature values is transferred from weather sensor nodes to the overlay weather application by applying data path. The control data like weather overlay commencement and weather overlay join request or reply operation is sent over the signaling path.

Fig.2. shows the nodes, layers and paths. There are three layers they are virtual weather sensor, physical weather sensor and weather application overlay and different paths are signaling path and data path. The wireless sensor network has two types of sensor nodes at the physical weather layer. Type A weather sensor node performs its own overlay weather

management activities and also for other weather sensor nodes, but type B weather sensor cannot. In figure 2 temperature sensor is a type A weather sensor node and pressure sensor and humidity sensor are type B weather sensor nodes. There is another network at the same layer, called the Gates-to-Overlay (GTO) network, comprising of heterogeneous nodes such as dynamic weather sensors, gateways and sink nodes. GTO nodes can interact with the WSN weather sensor nodes and help them also to join the weather application overlays. In this architecture, type B weather sensors have 2 options for joining the weather application overlays, either with the help of type A weather sensor nodes or GTO nodes. In fig.2, temperature sensor can perform overlay weather management operations for itself and for humidity sensor, whereas pressure sensor uses a GTO node to join the weather application overlay.

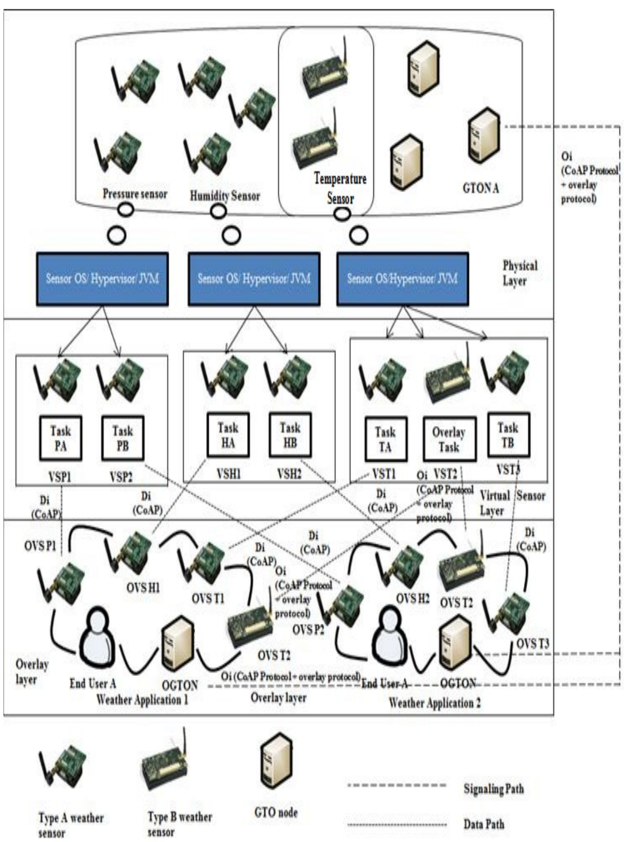

Fig.2. WSN Virtualization applied for Weather applications

The virtual weather sensor layer comprises of the virtual weather sensors that perform overlay weather application tasks or overlay weather management tasks. The virtual weather sensors of pressure sensor and humidity sensor only execute overlay weather application tasks, as they are type B weather sensor nodes. Temperature sensor, a type A weather sensor node, has 3 virtual sensors, 2 for the overlay weather application tasks and 1 (VST2) for the overlay weather management task. Both humidity sensor and temperature sensor use VST2 to participate in weather application overlays. The weather overlay layer consists of multiple application-specific weather overlays (for easy understanding only two overlays are shown). Each weather application overlay is designed by the weather analyst application and consists of two types of nodes, virtual sensors that run overlay weather application tasks and virtual sensor/GTO nodes that run overlay weather management tasks. In these overlays the boundaries brought by the physical weather WSNs disappear, allowing easier data exchange between overlay. As per the fourth principle, there are distinct paths between entities. The rules and interface used at these paths are the data path that uses the data interface provided by all the weather sensor nodes. CoAP [18] is a candidate protocol for this interface. The Gate-to-overlay interface is given by all-weather sensors as well as GTO nodes.

V. ID3 Algorithm Applied To Weather Data

ID3 is a non incremental algorithm that means it derives from known set of training instances. By considering a small set of training instances ID3 works on all future instances. The basic steps of ID3 algorithm [19] are choosing attributes, information gain, entropy, splitting criterion, rules to classify and information learning. Example ID3 algorithm applied for weather data is given in fig.3.

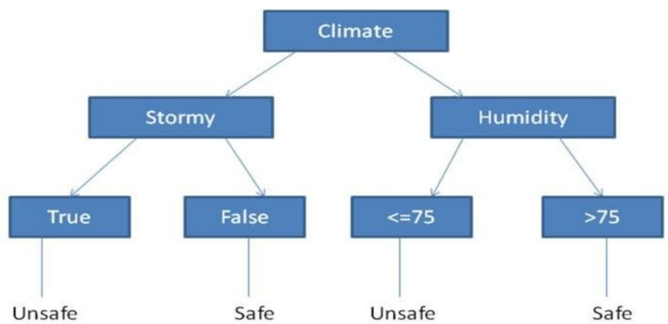

Fig 3: Example of ID3 applied on weather data

Entropy is known as level of uncertainty and highest information Gain is known as the criteria used for the splitting of weather data attributes for future prediction. ID3 entropy and Gain formulae used on weather data set is given.

$$\textbf{Entropy(S)} = \sum -p(I) \log_2 p(I)$$

$$\textbf{Gain(S, A)} = \textbf{Entropy(S)} - \sum ((|S_v| / |S|) * \textbf{Entropy}(S_v))$$

VI. Expermential Results

Our main focus in this paper is to present a new concept framework for weather data analysis and to send disaster alerts to SaaS users. A few experimental results so far consist of front end developed in HTML5 deployed on C-DAC scientific cloud. Sensed data parameters values can be retrieved from cloud on daily and monthly basis. For Example Fig.4, Temperature sensor data is retrieved for a particular day by entering year, month and date as parameters. This can be extended to retrieve data for all the weather sensor parameters and also to retrieve sensor data between a particular time period like 15 days data, 7 days data etc.

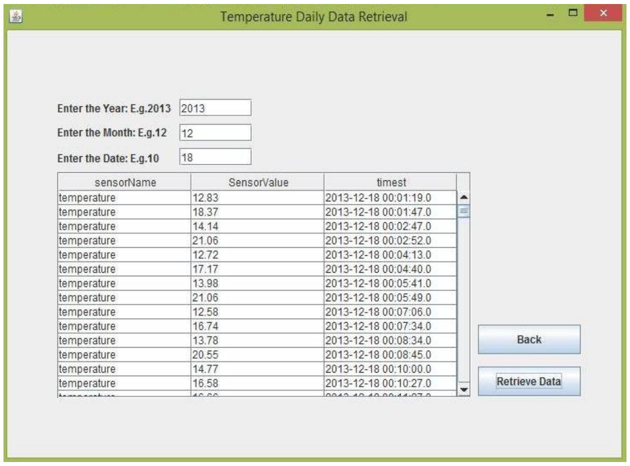

Fig.4. Temperature sensor data retrieved based on a particular time stamp.

Statistical analysis of temperature, humidity and other weather parameters are performed to easily extract and understand useful information such as maximum, minimum and average weather parameter values reached on a particular day useful for weather analysts to understand weather patterns. Statistical temperature graph given in fig.5 x-axis specifies timestamp. Where, every minute represents one unit on the graph and y-axis specifies sensor temperature value reached at that point of time.

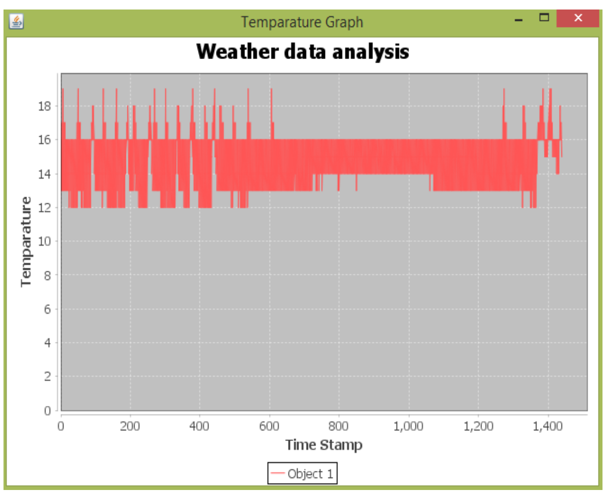

Fig.5 Statistical Analysis of temperature x-axis temperature value versus y-axis time stamp.

The statistical humidity graph is shown in Fig.6 x-axis specifies timestamp every minute is one unit and y-axis specifies humidity sensor value reached at that point. Humidity parameter plays a decisive role to determine information gain which is a major catalyst for making decisions and predicting weather in ID3 algorithm.

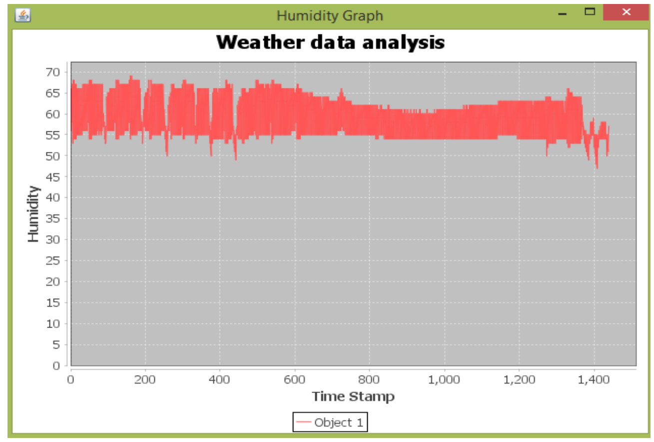

Fig.6 Statistical Analysis of humidity x- axis humidity value versus y-axis time stamp.

Weather activity detection engine is used to generate basic prediction alerts. When ID3 algorithm exceeds certain threshold a warning is send to weather analysts for further analysis. We have developed a stand-alone basic decision prediction ID3 algorithm shown in fig.7. Using few sensor parameters like humidity, temperature for decision making by inputting climate type, year, month, date and if the weather is stormy or not.

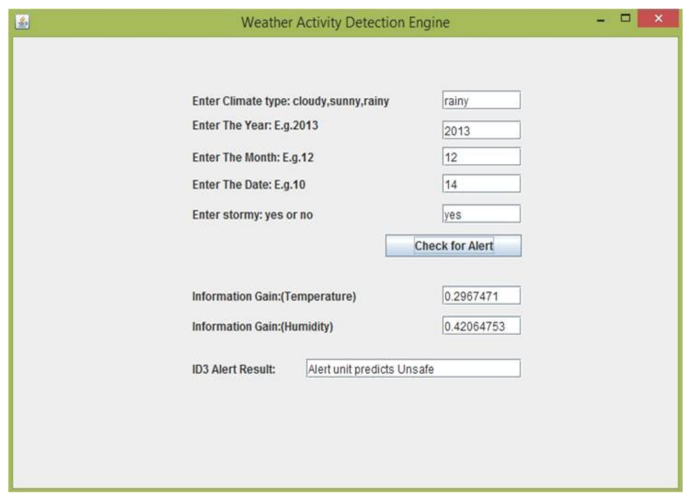

Fig.7 Weather Activity Detection Engine based on ID3 Algorithm

VII. Conclusion And Future Work

This paper focuses on the development of Cloud-WSN integrated SaaS framework to notify alerts to users during weather disasters so users can stay away from disaster sites. Node and network level virtualization is applied to avoid redundant WSN deployment of multiple sensors and also to support multiple weather applications for proliferation of weather sensors. Developing devices and tools to display and generate weather/disaster alerts using ID3 algorithm to SaaS users and weather analysts via SMS, emails which helps users divert to safer location. Cloud authentication is provided using the SSH and uses x-server for access control protocols. Our approach monitors weather data, daily weather activities and weather disaster activities at low cost. Cost is reduced since redundant sensors are not deployed for multiple weather application and sensor data storage, retrieval and analysis is performed inside a cloud. Authorized users have admittance to cloud data to view weather behaviors through web interface application. The web application also consists of monitoring of users and metering services on the application (pay-per-use). Development cost of this application is very low compared to previous techniques and can be deployed across cities and countries around the world with limited WSN resources to store and analyze weather data and send SaaS alerts to users during weather disasters.

Work so far is deployment of sensors based on node level virtualization and generating daily and weekly statistical graphs using algorithms. Future work includes deployment of network virtualization and cameras, implementing ID3 algorithm for all weather parameters to predict weather on cloud environment and develop mechanisms to integrate weather data from multiple sites that would be useful when sending SaaS disaster alerts to users.

ACKNOWLEDGEMENT

I would like to use this wonderful opportunity to thank my parents Subramanyam D S and Anitha D S, sister Archana D S and my uncle Shashi Kumar G S for their constant support and motivation. I would also like to convey my thanks to C-DAC, Bangalore for providing internship opportunity and also Department of CSE, RNSIT, Bangalore. I would like to extend my thanks to my internal guides at C-DAC Mr. Dwarakanath V and Mr. Arunachalam Bala and also my internal guide at RNSIT Mrs. Sampada K S for their guidance and pushing me to newer levels. I would also like to thank C-DAC Director Mr. N. Sarat Chandra Babu, Joint director Mr. Prahlada Rao B B and Principal technical officer Mr. Haribabu Pasupuleti for providing valuable inputs and also HOD of RNSIT College, Mr. G T Raju for his constant encouragement and lastly my beloved friends.

REFERENCES

[1] S.W. Smith and P. McDaniel, "Outlook: Cloud Computing with a Chance of Security Challenges and Improvements," IEEE computer and Reliability Societies, Jan 2010, pp. 77-80.

[2] Jamshed khan, Fazle Kabeer Khan , Wajid Ali, Sajjad Hussain shah, "A New Framework to Integrate wireless sensor networks with cloud computing," 2013, IEEE Aerospace Conference.

[3] Plamena Zlateva , Dimiter, Velev "A Feasibility Analysis of Emergency Management with Cloud Computing Integration," April 2012, International Journal of Innovation, Management and Technology, No.2, vol .3.

[4] Cholatip Yawut and Sathapath Kilaso "A Wireless Sensor Network for Weather and Disaster Alarm Systems," International Conference on information and Electronics Engineering, 2011.

[5] Rajkumar Buyya and Chen-khong tham "SensorGrid: Integrating Sensor Networks and Grid Computing," IEEE Technical Committee on Scalable Computing, 2005.

[6] Pei-Shan Yu , Wen-Yaw Chung, Chao-Jen Huang " Cloud computing system based on wireless sensor network," (FedCSIS), Federated Conference on, 2013.

[7] J. A. Nazabal, F. Falcone, C. Fernandez- Valdivielso, S. C. Mukhopadhyay and I. R. Matias "Accessing KNX Devices using USB/KNX Interfaces for Remote Monitoring and Storing Sensor Data," International Journal of Smart Home Vol. 7, No. 2, March, 2013.

[8] http://en.wikipedia.org/wiki/RRDtool.

[9] http://telegraph.cs.berkeley.edu/tinydb/doc/

[10] Sungyoung Lee, Xuan Hung Le, Phan Truc, La The Vinh "Secured WSN-Integrated cloud computing for u-life care," Consumer communications and networking conference (CCNC), 7th IEEE, 2010.

[11] Khan, I. "Design and analysis of virtualization framework for Wireless Sensor Networks," World of Wireless, Mobile and Multimedia Networks (WoWMoM) IEEE, 2013.

[12] Khandakar Ahmed, Mark Gregory "Integrating Wireless Sensor networks with cloud computing," Mobile Ad-hoc and Sensor Networks (MSN), Seventh International Conference on IEEE, 2011.

[13] Prahlada Rao BB, Payal Saluja, Ankit Mittal Neetu Sharma, Shivay Veer Sharma., "Cloud Computing for Internet of Things & Sensing Based Applications", 6th International Conference on Sensing Technology (ICST 2012), Kolkata, India. 18-21st December 2012.

[14] Payal Saluja, Prahlada Rao, Ankit Mittal, Rameez Ahmad, "CDAC Scientific Cloud: On Demand Provisioning of Resources for Scientific Applications", Proceedings of PDPTA'12: Conference on Parallel and Distributed Processing, Las Vegas, Nevada, USA , 16-19 July 2012.

[15] Deepansu S., Aman A., Ramesh Naidu L., Arunachalam B., Vineeth Simon A., Prahlada Rao B.B, Problem Solving Environment for Seasonal Forecast Model on CDAC Scientific Cloud, Second International Conference on Advances in Cloud Computing (ACC 2013), Conducted by CSI Bangalore, 19-20th Sept 2013.

[16] Fatna Belqasmi, Imran khan, Noel Crespi and Roch Glitho "A Multi-Layer Architecture for Wireless Sensor network Virtualization," Wireless and Mobile Networking Concference(WMNC), 6th Joint IFIP23-25, IEEE Apriil,2013.

[17] Lua, Eng Keong, et al. "A survey and comparison of peer-to-peer overlay network schemes," IEEE Communications Surveys and Tutorials 7.2, 72-93, 2005.

[18] Shelby, et al, "Constraint Application Protocol (CoAP)", IETF, Internet-Draft, draft-ietf-core-coap-13.txt (work in progress), 2012.

[19] Sonia Vatta, Rupali Bhardwaj et al. "Implementation of ID3 Algorithm," International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 6, June 2013.