International Journal of Scientific Research and Engineering Development-– Volume 3 Issue 3, May – June 2020

Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

Fog Node Possible Architectures and Propose the Changes in Mechanism to Architecture to Make Contiki OS Real Time

Dharmendrasinh .J Revar[1], Tejalba R. Talatiya[2] , Raj Hakani[3], Hemal D. Nayak[4]
[1,2]PG Student,[3,4]Asst. Prof.
GTU-Graduate School of Engineering and Technology,
Sector-26, Gandhinagar, Gujarat, India

Abstract:

Fog computing has become emerged as a promising technology that can bring the cloud applications near to the physical IoT devices at the network edge. It is widely known what cloud computing is, and how the data centers can build the cloud infrastructure and how applications can make use of this infrastructure, there is no common picture on what fog computing and a fog node, as its main building blocks. IoT has contiki OS that can communicate with iot devices and send the data on the cloud but its not a fully real time and fast. This paper gives a perfect overview of the fog node architecture and its challenges.we propose a change in the scheduling mechanism in the fog so that make the contiki IOT os communicate with real time. for making to communicate IOT devices real time i propose priority wise and round robin algorithm in scheduling mechanism on the fog node.

Keywords: IoT, Fog node, Contiki

INTRODUCTION: IOT(internet of Things) is a technology that can connect all the devices to the cloud And devices are controlled and monitor through cloud.After IOT is coming in to the Picture the sensor are attached with device that can give the values that can be measured from any were in the world through cloud by accessing remotely.The IOT Has been used in many sectors like smart home,smart city,smart hospitals,smart Television and Smart Phones etc.IOT has reduced the cost of monitoring the patients Through cheap sensors that can be helpful for doctor as well as patients.The use of IOT devices has been increased rapidly day by day.IOT can change the way of our Living.

Software Platforms, Which facilitate design and implementations of IoT applications may bring more developers into this domain.For the sake of increasing need for application development on IoT, several operating systems for Wireless Sensor Network (WSN) and IoT motes are realized. contiki an Tiny OS best known operating systems for wireless sensor networks.Fog computing is introduce to reduce the data load to the cloud.in iot OS more and more sensors constantly sends the data to the cloud. This can make lots of load to the cloud and also take more Time. The cloud has limited storage so inorder to send all the data to the cloud we only want to important and data to the cloud.so that fog computing concept was introduce. In Fog computing the computation is become at the edge of devices. so that most of processing is done on the edge of the devices this can make computing fast flexible and reliable.

CONTIKI OS:Contiki is widely used operating system for Internet Of Things (IOT). It is capable for low power low memory devices and sensors. Such applications for Contiki include systems for street lighting, sound monitoring for smart cities, radiation monitoring, weather monitoring and alarms. Contiki is an open source softwere and it is Release Through BSD licence.

The first creator of contiki is Adam Dunkels later it is developed by team of Texas instruments, Cisco, zolerita and many others.

Contiki become popular because its have built in TCP/IP stack and also its lightweight Preemptive scheduling on event-driven Kernel.

Contiki has 10 Kilobytes of Random access memory and 30 kilobytes real only memory and it is a full Graphical user interface system it need only 30kb ROM.Recently the developers created new branch, it is known as Contiki-NG. This operating system For next generation IOT

The running Contiki system have four parts as per below:

1. Kernel
2. Libraries
3. Program loader
4. Set of processes(it can be application program or service)

The processes are defined by its even handler function and an optional poll handler function. The state of the process is done in its own memory though the kernel only holds a pointer to the process state. All processes shared same address space,they all do not run in different protection(secure) domains. Later Inter-process communication (IPC) is done by posting events.

As shown in Fig.1 below, the Contiki system is divided into core and loaded programs, and this division is made at compile time. The core has inbuilt Contiki kernel, the program loader, the most commonly used parts of language run time and support libraries.

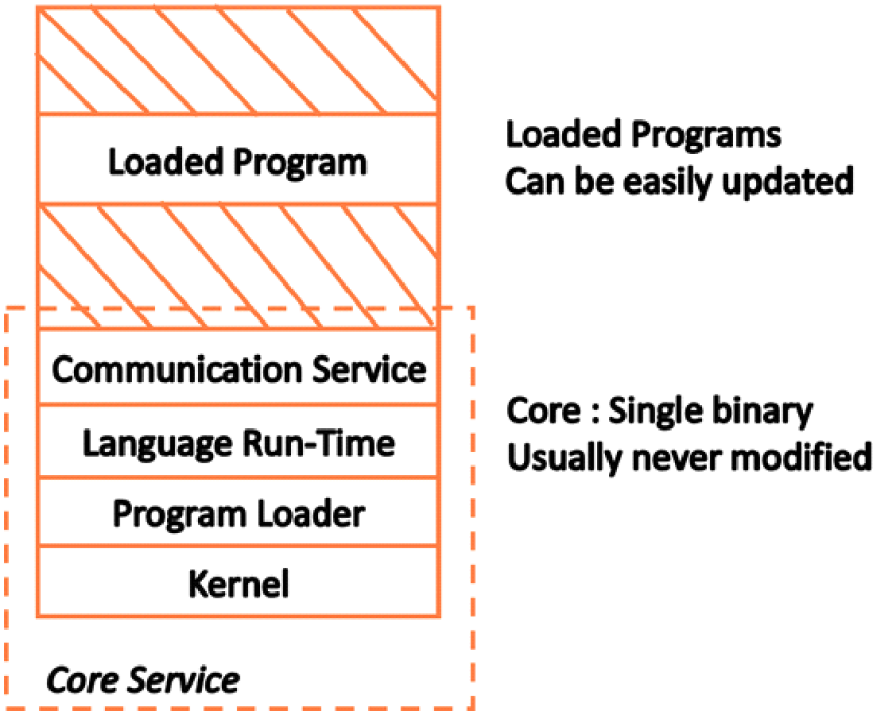

Fig.1 Contiki OS Architecture

FOG NODE:

The Fog Computing Is An Emerging Technology today. The fog node is near to the devices. It can be act as a intermediate between cloud and edge devices.

How different features of edge devices can be presented as logical way in a fog node is an open question. Also, what the computing entity in the all system is,or its locality, where these abstractions are created and controlled is open to discussion.

The physical devices which building the fog node (preferably the one with higher computing capacity)can be made responsible to deploy the abstraction, same as the concept of cluster lead. while it can also granting communication between all fog layers and the cloud. In a more appropriate way of today's systems, this could be refereed to as the fog node controller.

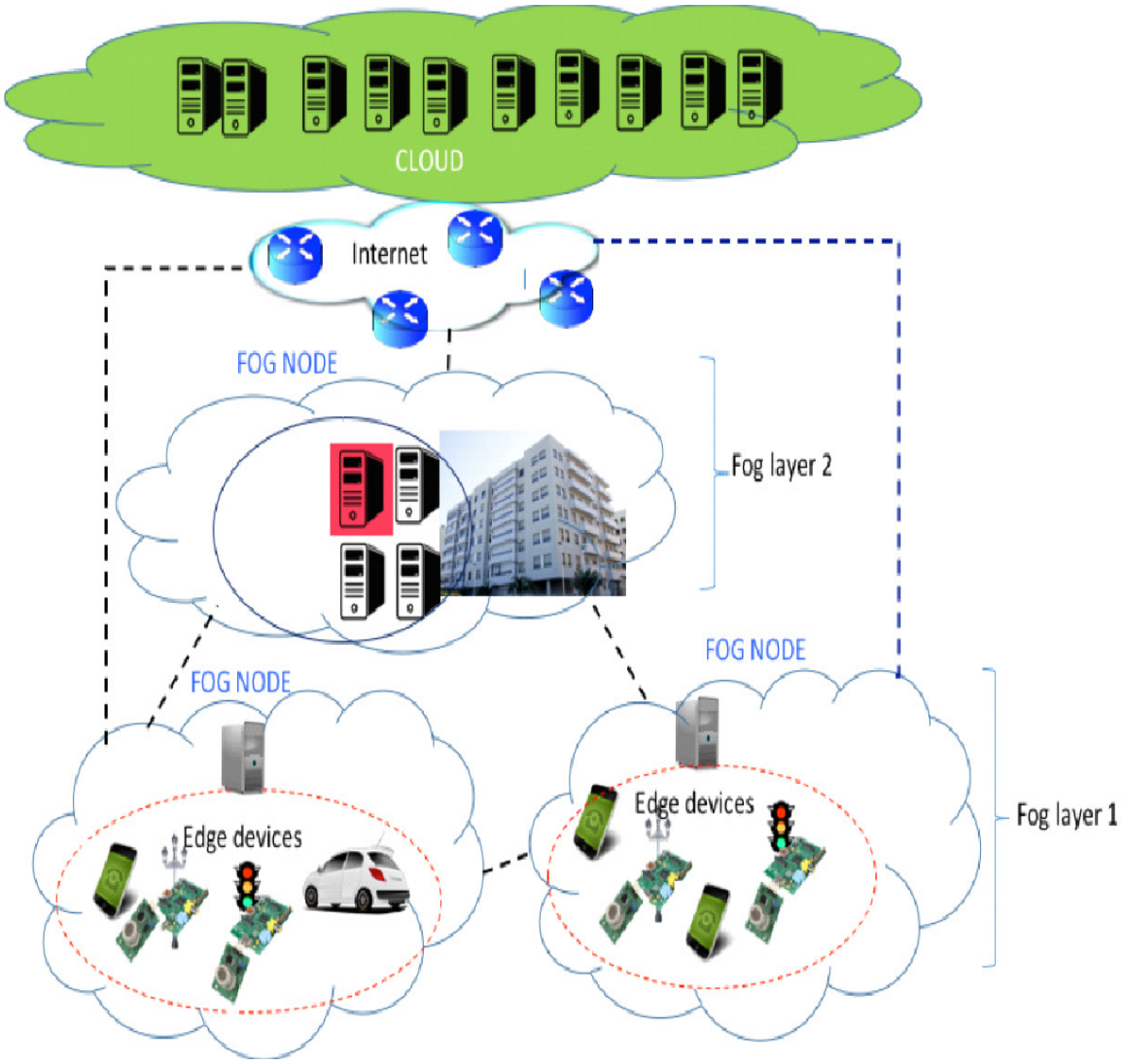

Fig.2 Fog to Cloud Architecture

Fog nodes are distributed fog computing entities that can enabling the deployment of fog services, and formed by at least one or more physical devices with processing and sensing capabilities (e.g., computer, mobile phone, sensors, etc.).
Different physical devices of a fog node are connected by different network technologies wired or wireless and aggregated and abstracted to be viewed as one single logical entity, that is the fog node, able to execute distributed services, as it were on a single device.

FOG ARCHITECTURE: Fog architecture has been proposed by many developers. Fog node is the nearest node of iot devices. In most cases, time-sensitive data was analyzed on the fog node closest to the things generating the data.

The centralized aggregation cluster/node is used for analysis of data that can be wait for seconds or minutes and then according to it action is performed. The data is less time sensitive For historical analysis and storage purpose,has been sent to the cloud. for example here would be each of the fog nodes sending periodic summaries of data to the cloud for historical analysis.

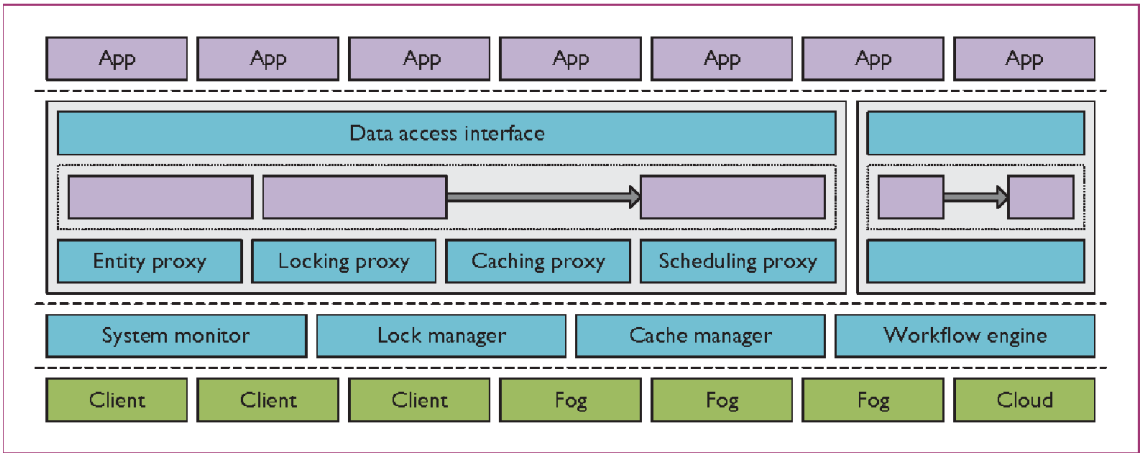

Figure 3. WM-FOG software stack.

The top layer is called application layer, where user applications has inside. The next layer is the work flow layer, where work flow instances has inside. Under the work flow layer is the system layer, where the system components has inside. The bottom layer is the entity layer, where all the system entities like client devices, fog nodes, and the cloud has inside.

Limitation of contiki OS:

contiki OS have limitation it has not been real time.

Round Robin scheduling

Round Robin is CPU scheduling algorithm in which every process has been assigned a fixed time slot in a cyclic way.

- It is simple, easy to implement, and starvation-free as all processes get fair share of CPU.
- One of the most commonly used technique in CPU scheduling as a core.
- It is preemptive as processes are assigned CPU only for a fixed slice of time at most.
- The disadvantage of it is more overhead of context switching.

ROUND ROBIN EXAMPLE:

Process	Duration	Order	Arrival Time
P1	3	1	0
P2	4	2	0
P3	3	3	0

Suppose time quantum is 1 unit.

P1	P2	P3	P1	P2	P3	P1	P2	P3	P2

0 10

P1 waiting time : 4

P2 waiting time: 6

P3 waiting time: 6

The average waiting time(AWT) : (4+6+6)/3=5.33

Illustration: Completion Time: Time at which process completes its execution.

Turn Around Time: Time Difference between completion time and arrival time.

Propose Solution:

Turn Around Time = Completion Time – Arrival Time

Waiting Time(W.T): Time Difference between turn around time and burst time.

Waiting Time = Turn Around Time – Burst Time

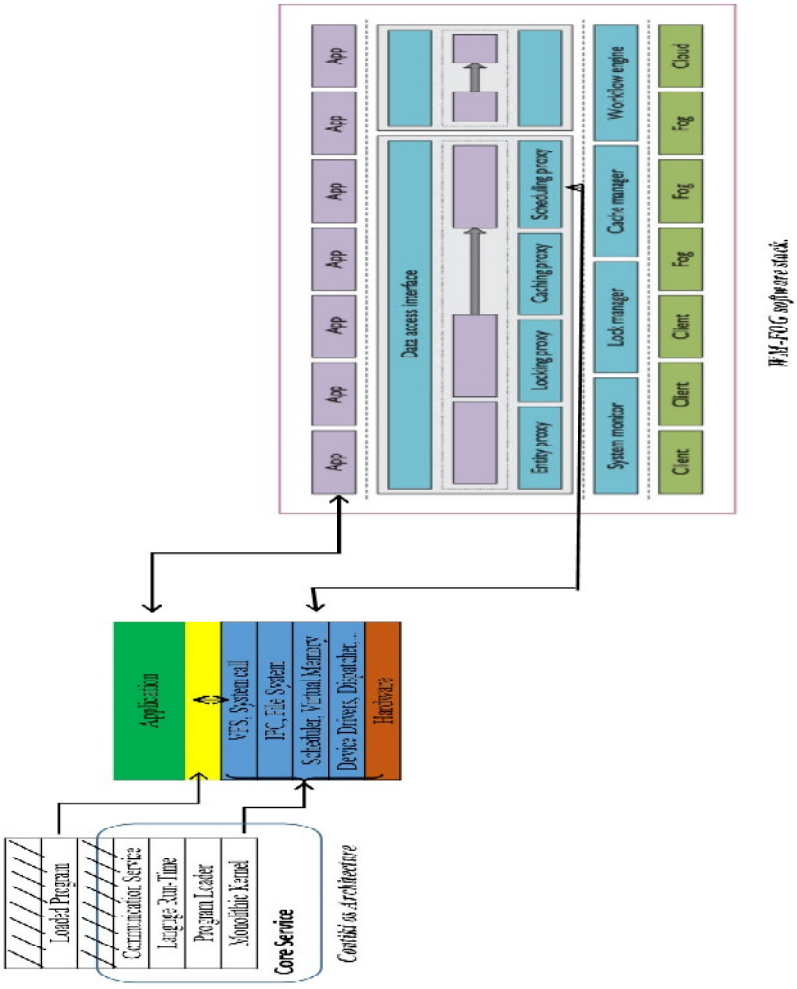

Fig 4. Proposed Architecture

we propose priority wise round robin scheduling inside the proxy scheduling in the architecture Fog computing. So that it can make Contiki OS real time scheduling. So benefit we get real time data in the cloud and it is more reliable.

CONCLUSION – Fog node has many possible architectures. Out of which Contiki OS can be worked with real time fog node to provide real time interactions. This purpose will be satisfy using the priority based round robin algorithm that will enhance the uses of fog nodes and efficiency of fog node also will increase. Which can be prove tremendous advantages in near future using Contiki OS and fog computing.

REFERENCES:

1]Zijiang Hao, Ed Novak, Shanhe Yi, Qun Li. "Challenges and Software Architecture for Fog Computing", IEEE Internet Computing, 2017

2]V. Ceronmani Sharmila, A. George. "Chapter 1 Implementation of Connected Dominating Set in Fog Computing Using Knowledge-Upgraded IoT Devices", Springer Science and Business Media

3]Tammishetty S, Ragunathan T, Battula SK, et al. IOT-Based Traffific Signal Control Technique for Helping Emergency Vehicles. In: Proceedings of the First International Conference on Computational Intelligence and Informatics. Springer; 2017. p. 433–440.

[4] Yousefpour A, Ishigaki G, Gour R, et al. On Reducing IoT Service Delay via Fog Offlfloading. IEEE Internet of Things Journal. 2018;PP:1–12.

[5] Aazam M, Huh EN. Fog computing micro datacenter based dynamic resource estimation and pricing model for IoT. In: Proceeding of the 29th IEEE International Conference on Advanced Information Networking and Applications (AINA). IEEE; 2015. p. 687–694.

[6] Arkian HR, Diyanat A, Pourkhalili A. MIST: Fog-based data analytics scheme with cost-effificient resource provisioning for IoT crowdsensing applications. Journal of Network and Computer Applications. 2017;82:152–165.

