ISSN: 2278 – 7798
International Journal of Science, Engineering and Technology Research (IJSETR)
Volume 5, Issue 5, May 2016

A Detail Review on Cloud, Fog and Dew Computing

Pooja Kukreja[1], **Dr. Deepti Sharma**[2],

M-Tech Student, Department of CSE, Advance Institute of Technology and Mgt, Palwal, Haryana, India[1]
HOD, Department of CSE, Advance Institute of Technology and Mgt. Palwal, Haryana, India[2]

ABSTRACT
The paper takes the conceptual method for organization of the vertical hierarchical connections between the scalable distributed computing paradigms: Fog computing, Cloud Computing and Dew Computing. In this paper, the Dew Computing is explained and realized as a novel structural layer in the available distributed computing hierarchy. In the available computing hierarchy, the Dew computing is placed as the base level for the Fog and Cloud computing paradigms. Vertical, hierarchical and complementary division from Cloud to Dew Computing fulfills the requirements of low and high end computing needs in everyday work and life. These novel computing paradigms decreases the cost and enhance the performance, specifically for applications and concepts i.e. Internet of Everything (IoE) and the Internet of Things (IoT). In summation, the Dew computing paradigm will need novel programming models that will effectively decrease the complexity and enhance the usability and productivity of scalable distributed computing, adopting the High-Productivity computing principles.

I. INTRODUCTION

Cloud computing networks are huge groups of servers and cloud service providers that normally take benefit of low-cost computing technique, with particular links to distribute data-processing chores across them. This shared IT infrastructure consist large pools of systems that are connected together [3]. Virtualization methods are usually utilized to increase the cloud computing power. In cloud computing, the word cloud (also called as "the cloud") is utilized as a metaphor for "*the Internet*," so the phrase *cloud computing* implies "a kind of Internet-based computing," where several facilities i.e. storage, servers and applications are provided to an organization's devices and computers across the Internet [11]. This architectural design introduces many new characteristics, i.e. resource sharing and high computational abilities. High computational abilities are obtained by integrating together a large no. of compute units through a fast network, while resource sharing permits various distributed entities to be shared between different subscribers depending on the user's needs and resource availability. Furthermore, adding, deleting and accessing the resource is easy and can be performed uniformly, permitting many devices to communicate and share data among themselves.

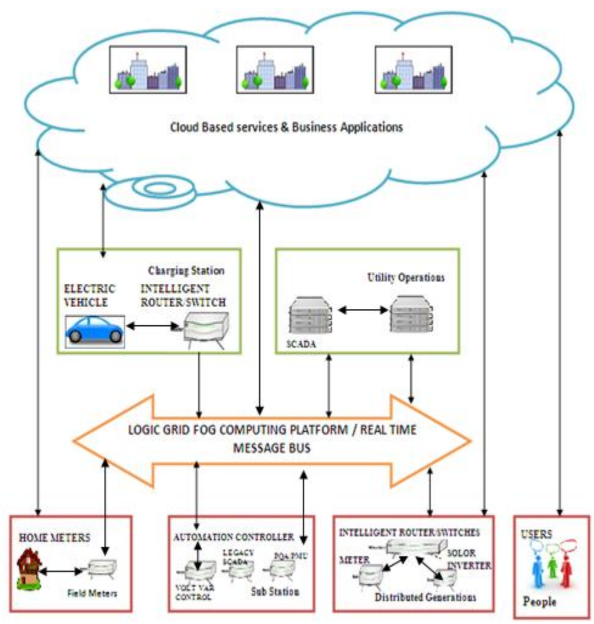


FIGURE 1:-FOG AND CLOUD ARCHITECTURE

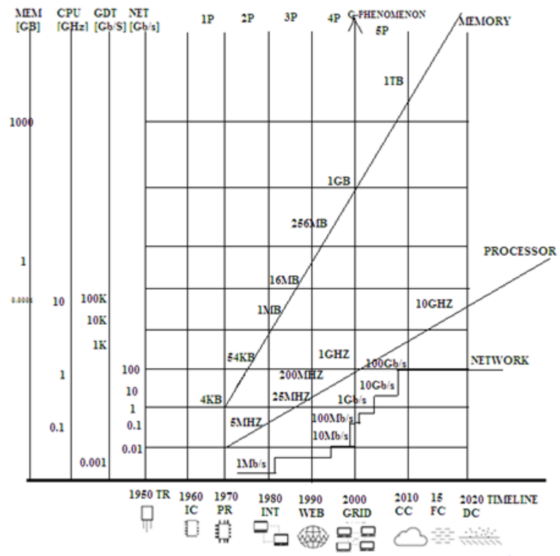
The largest boost in the growth of distributed computing happened around year 2000 when the network bandwidth, processor clock rate and RAM (Random-Access Memory) capacity arrived the Giga range. This happening has been represented as the G-phenomena [21] and began the fifth phase of development of computing systems (view Figure 1).

Particularly, 1 GHz processors were issued by AMD and Intel, gigabit Ethernet was in use and the first computers with 1 GB of RAM became existed. This alignment made a virtual combination of spatially distributed computers believable, further making enable quick growth of distributed systems, hence producing situations for space-independent conception of distributed systems.

Began by the Cloud and Grid computing paradigms, the G-phenomena was one of the two primarily driving forces (i.e. software and hardware) that lead to the current growth of Information and Communications Technologies (ICT). The main susceptibility for that growth was obtaining substantial speed enhancements of processors and their inter links, and the capability to process large amount of data in memory. High-performance distributed computing systems were discovered on Grid computing paradigm while scalable distributed computing systems developed via Internet of Things and Cloud computing paradigm. Similarly, in several branches of modern research i.e. climate change simulations, genomics, medical research, drugs discovery, data intensive and computational problem have produced. These problems cover the creation, analysis and interpretation of large amounts of data and their comparisons against catalogs of available knowledge via complicated multistage workflows. These analyses or workflows are enabled by a integration of analysis platforms and computational infrastructures and can be offered as three important Cloud services: Platform as a Service (PaaS), Infrastructusre as a Service (IaaS) and Software as a Service (SaaS). Since, for satisfying the requirements of the current as well as future research problems, for instance real time human brain simulation, one of the primary issue in the computer science, the scalability, has to be figured out. It is required that the future computing systems will be highly distributed, composed of several devices i.e. conventional computers, mobile devices, embedded systems and sensors, that the present-day paradigms and models could not effectively solve. Hence, novel challenges will need new computing paradigms and models. As an instance, a relatively novel computing model that can effectively solve data-intensive problems with decreased power consumption depend on the Field Programmable Gate Arrays (FPGAs) employing a data-flow method to processing [13].

II. HIERARCHY OF SCALABLE DISTRIBUTED COMPUTING

Scalability is the feature of a computer system, application or network to deal with a developing work, both in terms of storage resources as well as processing power, or its possibility to be easily enlarged for accommodating that development. Today, we are seeing the exponential development of data (Big Data) and processing (application) requirements which cause to the essential resources scalability at various levels. In this manner we came to a novel hierarchical structure, composing of three



G(GIGA)-PHENOMENA,TR-TRANSISTOR,IC-INTEGRATED CIRCUIT,PR-PROCESSOR,INT-INTERNET,BD-BIG DATA,CC-CLOUD COMPUTING,FG-FOG COMPUTING,DC-DEW COMPUTING,1P....5P-DEVELOPMENT PHASES.

Figure 2: The decade phases of the development of ICT and its relations to the G-phenomena and Distributed Computing Hierarchy: Cloud Computing (CC), Fog Computing (FC) and Dew Computing (DC).

Layers: Fog, Cloud and Dew Computing [20]; scalable platforms that show a novel computing paradigm, as explained in Fig 3. These hierarchical layers are carried out to provide the rapidly growing complex distributed computer systems and satisfy the following needs:

Performance: analyzed for fast processing, responses and less latency;

Availability: needs redundancy, fast recovery in the situation of system failures, and graceful reduction when problems happen;

Reliability: system requires to be reliable in function and data;

Manageability: scalable system that requires little effort to operate; Cost: involve software and hardware costs but it is also significant to consider other aspects required to deploy and manage a scalable computing system.

Figure 3 illustrates the hierarchical division of CC / FC / DC with the resource, structural and application facets. Vertical growth of scalable virtualization in a

ISSN: 2278 – 7798
International Journal of Science, Engineering and Technology Research (IJSETR)
Volume 5, Issue 5, May 2016

distributed computing system has a part to explore the application domain to the largeness of applications through a variety of categories and types in the connection chain integrity and overall application services.

2.1 Cloud Computing

The term Cloud Computing (CC) was first proposed by Professor Ramnath Chellappa in 1997 in a lecture in which he explained it as a novel computing paradigm where the computing boundaries will be found by economic rationale instead of technical restrictions [9]. Cloud computing enables effective maintenance of data centers, virtualization of resources and time-sharing with a particular stress on the business model [3][4][8].

By this paradigm, businesses and subscribers are capable to have Platform-as-a- Service Infrastructure-as-a-Service and Software-as-a-Service models on need from anywhere in the world. When Cloud's facilities are made existed by cloud supplier in a pay-as-you go way to the general public, it is called as Public Cloud, while Private Cloud means internal Cloud platforms possessed and used only by specific organizations. Different from Grids, Clouds can scale up or down as per the needs of subscribers and at the same time enable more elastic and isolated execution environments for applications.

In Grid computing the logical alternative was to hold the

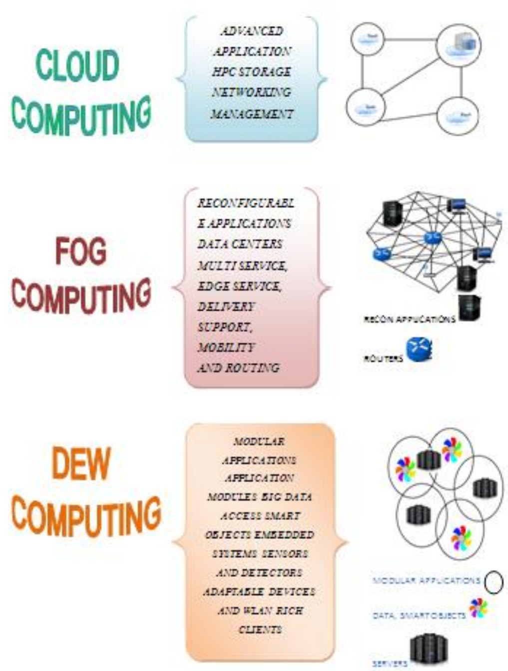


Figure 3: Scalable Distributed Computing Hierarchy.

Heterogeneousness of the computing equipment to the least possible (same processor type, operating system, software libraries etc., specifically at a particular Grid site, but also across the whole Grid infrastructure), while the Cloud, through virtualization, permits much broader possibility of alternatives. From the end-users point of view, Cloud provides a service (IaaS, PaaS or SaaS) that conceals the underlying heterogeneity and complexity which is observed by the subscriber as a homogeneous, single service. That refers that the subscriber requirements will be processed by an already available software/hardware combination, with no inherent limitation in the CC paradigm, which would not allow (or minimum make hard) the utilization of very heterogeneous computing resource. The subscriber is not really known of the underlying hardware, for example, processor, operating system, disk or memory utilized by a specific part of the Cloud. With the development of the Cloud techniques more companies, particularly SMEs, are moving from self-maintained, self-acquired and self-provided SaaS and PaaS solutions to more elastic, time-sharing oriented Cloud solutions which provide the growth and build their business products and processes more competitive.

2.2 *Fog Computing*

The novel generation of smart devices, capable of processing data temporarily instead than forwarding them to the Cloud or any other computational system for processing, makes a new field of applicability and possibility for distributed systems. The Fog Computing (FC) [1] paradigm, first proposed by CISCO Systems, Inc., is a disseminated computing paradigm that offers data, storage, compute and application facilities nearer to client or near-subscriber edge devices i.e. network routers. Moreover, FC deals with data at the network level, on the end-user client side and on smart devices (for example mobile devices), rather than forwarding data to a remote place for processing. The objective of the FC is to enhance the efficiency by directly processing of data on the network (for example on smart devices or network routers) and hence decreasing the amount of data that has to be transmitted for processing but also holding data nearer to the subscriber hence improving the security which is major problem in CC [19]. The different features of FC are its closeness to particular end-subscribers, its dense geographical distribution, the capability to reconfigure itself and its potential as well as its support for mobility to increase security. Facilities in FC are hosted at the network layer, which decreases service latency and enhances the entire Quality of Service (QoS), resulting in a higher level of distributed computing services. Today, with the large amount of sensors and devices that produce high amounts of data, the FC paradigm is becoming progressively attractive and famous. Furthermore, in several applications, transmitting a high amount of data from sensors to farther processing systems can be ineffective and possibly cause to an unessential network overload that can reduce security and performance (for example: real-time vehicle-to-vehicle communication). Taking this example, the FC paradigm is well located for Big Data and knowledge extraction and real-time analytics. FC is a novel conceptual solution for covering the needs of the growing no. of Internet- oriented (or linked) devices sometimes known as the Internet of Things (IoT) [6][16]. In the IoT mechanism, the term thing means any artificial or natural data-oriented object that an Internet Protocol (IP) address can be allocated to, hence offering it with the capability to achieve and transfer data across a network to a remote (distributed) computing environment [7]. Furthermore, today, a large no. of Things linked to the Internet are able of generating large amounts of data, for which FC will play a main part in decreasing the latency in dynamical network-based applications.

2.3 *Dew Computing*

Dew Computing (DC) goes beyond the concept of a network/ service/storage, to a sub-platform - it depends on a micro-service idea in vertically distributed computing hierarchy. In comparison with Fog Computing, which provides support in developing IoT applications that need predictable and real time latency and the dynamic network re-configurability, DC forces the frontiers to computing applications, low level and data services away from centralized virtual nodes to the end subscribers

In [23] the writer introduces a cloud-dew kind of architecture that distributes the information around the end-subscribers devices (for example smart-phones or laptops) hence enabling data access in a situation even when no Internet connection is present. The DC scheme also provide benefits to resources that are continuously linked to a network, i.e. tablets, tablets and sensors. As a result, DC deals with a broad range of techniques involving mobile data acquisition, wireless sensor networks, cooperative applications on distributed peer-to-peer processing environments and ad hoc networking. These explore further to autonomic self-managed networks, augmented reality applications and data intensive applications that open each day, general purpose utilization in living and working environments, down to the level of the simple and particular purpose micro-services and equipments.

One of the primary benefits of the Dew Computing is in its greater scalability. The DC model depend on a large no. of different devices and different kinds of equipment, covering area from smart phones to intelligent sensors, combined in peer-to-peer ad hoc virtual processing environments composing of various micro services. Though highly heterogeneous, the active equipment in DC is capable to perform complicated tasks and efficiently run a wide variety of tools and applications. For providing such services, the devices in DC are self-adaptive and ad hoc programmable, able of running applications in a distributed way without needing a central communication point (for example: a central or master device).

III. LITERATURE REVIEW

Krishna tej Koganti et al. (2013) In this paper the author has evaluated that in order to share the data andone of the important technology in the cloud computing is virtualization. The main objective of the virtualization is its ability to run the multiple operating systems on a single machine buy sharing all the resources that belong to the hardware. In this paper the author has provided the basic knowledge about the virtualization technology in cloud computing and how it acts in the cloud computing environment. And we also discuss about, how to maintain the virtualized environment with the optimized resources. The author has said that the basic virtualization technology consists of cloud users and the service models, operating system and the hardware and has discussed the pros and cons of both traditional and virtual servers. And has also described the effects which arise when implementing the virtualization technology in cloud computing environment. Finally the author has concluded the paper by saying that by using the virtual mechanism we can take the cloud computing technology to another step forward in all aspects. So, by using the virtualization technology it leads to efficient usage of cloud computing environment.

Swati Aggarwal et al.(2016). The author has discussed that Cloud computing is a model of sharing computing resources over any communication network by using virtualization and it offers many benefits, one of them is elastic resource allocation. To fulfill the requirements of clients, cloud environment should be flexible in nature and can be achieve by efficient resource allocation. Elastic resource allocation is required to optimize the allocation of resources, minimizing the response time and maximizing the throughput to improve the performance of cloud computing. Fog computing is the virtualized intermediate layer between clients and cloud. It is a highly virtualized technology which is

similar to cloud and provide data, computation, storage, and networking services between end users and cloud servers. This paper presents an efficient architecture and algorithm for resources provisioning in fog computing environment by using virtualization technique.
The author has served various existing algorithms related to optimal resource allocation and different scheduling techniques. In this paper, the author has proposed and implemented an efficient resource allocation architecture and algorithm (ERA) on cloud analyst tool to test the performance of the proposed technique in the fog environment. By implementing the proposed strategy, the author has stated that the proposed strategy can be allocated resources in optimized way and better than existing algorithms in terms of overall response time, data transfer cost and bandwidth utilization in fog computing environment. This paper shows the comparison of existing resource allocation strategy with the proposed algorithm in terms of overall estimated response time and cost. This proposed algorithm can effectively minimize the overall response time, data center processing time and total cost. So, proposed strategy fully utilizes the bandwidth of the system, provide effective response time, data processing time and reduce data traffic over the internet in fog computing environment to allocate the resources.

IV. FOG SIMULATION TOOLS

Simulation software depends on mechanism of real phenomenon with a collection of mathematical formulae's. This software offers the simulated environment that is same as real world atmosphere. Simulation software is made in a way so that result should be near to real world. Continuous simulation and discrete event are the two classes of simulation package. Discrete simulations are utilized to simulate statistical events i.e. customer reaching in bank; while continuous simulation is utilized in broad variety of physical process i.e. human respiration, ballistic trajectory and radio frequency data communication etc. Cloud environment can also be modelled with the support of simulation software. Some cloud simulators that can be employed for cloud simulations are as:

1. iCanCloud
2. OPNET
3. CloudAnalyst

A. iCanCloud: - It is a modeling platform targeted to simulate and model cloud computing system. The primary aim of iCanCloud is to report the trade-off between cost and performance on a specific hardware. This tool is highly appropriate from basic subscribers to developers of huge distributed applications. The important characteristics of iCanCloud involve the following. 1. Available and non-available cloud computing architectures can be simulated and modeled. 2. Cloud broker schemes can be examined because of reliable hypervisor. 3. VM can model unit-core/multi-core systems. 4. A broad variety of storage options can be chosen ranging from remote storage systems like parallel file systems , Network file system (NFS) and redundant array of independent disk (RAID) systems and local storage systems. 5. Offers user friendly, easy graphical user interface (GUI) to simplify the generation and customization of huge distributed models. 6. Offers POSIX-based API and a compromised MPI, Library.7. Offers characteristic to add new component in iCanCloud repository to obtain more services.

B. OPNET: - OPNET's is normally specialized for network development and research. It is reliably utilized for communication networks study, about protocols, devices and the applications. As this is commercial service supplier it has a good graphical interface for subscriber and the graphical interface is utilized to make the network topology application and entities from the system application layer to the physical layer. Here, the object oriented programming language is utilized to generate a mapping from the graphical interface for the real implementation. The diagram below represents the graphical representation of every network nodes and the graphical output. As it has a graphical aspect, the parameter can be varied and seen the result repeatedly very easily without much hard work. This modeler is famous for network research and industry for the development. The provided programming tools and GUI interface are very useful to make the system according to subscriber need and to model the system. OPNET has three important services as modeling, simulation and analysis. For modeling it offers good graphical interface to describe and generate all types of model protocol. For simulation it utilized different kind of advanced simulation technique to deal and address broad range of study aim. For analysis, the simulation results and data can be showed graphically in user friendly forms of graphs, charts and in statistics form for subscriber comfort.

C. Cloud Analyst: - Cloud Analyst was obtained from CloudSim and explores some of its abilities and characteristics introduced. Cloud Analyst distinguishes the simulation experimentation exercise from a programming exercise. It also enables a simulator to repeatedly do simulations and to carry out a set of simulation experiments with little parameters variations in a easy and frequent way. Cloud Analyst can be used to check the large scaled Internet application behaviour in a cloud environment**.** The significant issue that subscriber comes across while performing with CloudSim is that it is not at all graphical in behaviour, so to have better results of visualisation, subscriber can choose for Cloud Analyst. This environment supports in distinguishing the simulation environment from the programming environment [5]. This simulation tool is very easy to employ and has the capability to generate the output in graphical form.

D. Cloud Sim: - CloudSim is a toolkit (library) for Cloud computing environments simulation evolved in the CLOUDS Laboratory at the Computer Science and Engineering Department of the University of Melbourne, Australia. CloudSim is generalized, new and extensible simulation toolkit that enables continuous simulation, modelling and experimentation of evolving cloud computing system, application and infrastructure environments for internetworked and single clouds. In simple words, CloudSim is a development toolkit for Cloud scenarios simulation. CloudSim is not a model as it does not offer a ready to utilize environment for execution of a whole scenario with a particular input. Instead, subscribers of CloudSim have to formulate the Cloud scenario it wants to measure, describe the needed output, and offer the input parameters CloudSim toolkit supports system and behaviour modelling of cloud system components i.e. virtual machines (VMs) data centers and resource provisioning schemes. It is significant to notice that CloudSim is not a ready-to-use solution where you adjust parameters and gather results for usage in your project. All the CloudSim components interact with each other via message passing.

E. Network Cloud: - Network Cloud is an extended version of CloudSim and is capable of enforcing network layer in CloudSim, reads a BRITE file and creates a topological network. Here, we have configuration file which consists the no. of nodes along with the several entities included in simulation [4].In this simulation tool, every entity is to be mapped to a single BRITE node so that network CloudSim can work appropriately. Network CloudSim can be utilized to model network traffic in CloudSim.

F. Network Simulator 2 (NS2): - NS2 is the most famous network simulators. This is a discrete event modeller mainly designed for the network researchers. NS2 is the second version of NS (Network Simulator) and NS was formulated in 1989. The latest version of NS2 is broadly utilized for academic research. After that lots of packages are contributed by some non-profit groups to enhance and build it much better.

Network Simulator or in short NS2 is an object oriented discrete event driven network modeller. It was first formulated at the California-Berkely University. The programming language utilized is Tcl script and C++ language with object-oriented extension (OTcl). There is cause utilizing these two languages. C++ is very effective to design but complicated for visual and graphical implementation. OTcl is employed to fill the lap that the C++ lacks. So the integration of these two languages appears to be very efficient. Normally the C++ is employed to implement the detail simulation protocol and OTcl is employed for the subscriber to control the simulation and organize the events. The OTcl script is utilized to start the event scheduler, to establish the network configuration and to tell traffic source whether to forward or stop forwarding the packet from event scheduler. The view can easily change by the OTcl script. There is reliability that when a subscriber wish a new network object they can simply write the code utilizing the available object library and also plumb the data path from object.

G. OMNeT++: - OMNeT++ is a public source, a discrete event modeller with GUI support of component based network modeller. The primary application field of this simulator is the communication networks along with its reliable architecture it has other areas i.e. hardware architecture, IT systems, queuing network and also in business mechanism. Here the components are known as modules and programmed in C++ language. Its operating principal is same as that of Python in NS3 and OTcl in NS2. The smaller components are integrated into larger components and models utilizing high level language.

The OMNeT++ is intended particularly for the complex based architecture. Basically the reusable components are aggregated to build OMNeT++ module. The major characteristics of OMNeT++ are the modules are reusable and the modules are integrated in several ways. The key characteristic is the simulation kernel C++ class library which contains utility class and simulation kernel essential for simulation components. It has runtime subscriber environments and interface for simulation. OMNeT++ support multiple platform like it can operate on Unux, other Linux-like systems and on Windows systems.

H. Open Cirrus: - Open Cirrus is an open cloud computing testbed sponsored by Inter, Hewlett-Packard (HP)and Yahoo! in cooperation with some other organizations. In accordance of [6] the Open Cirrus aspires to obtain the below objectives: Foster systems-level research in cloud computing. promote new cloud computing applications and applications-level research. provide a set of experimental datasets. formulate APIs and open-source stacks for the cloud. Open Cirrus offers a cloud stack consisting of virtual and physical machines, and global facilities i.e. monitoring, sign-on, job submission and storage.

I. CDOSim: - CDOSim is a cloud deployment option (CDO) Simulator which can model the SLA violations, response times and costs of a CDO. A CDO is a decisions relating simulator which takes decision about the choice of a specific runtime adaptation strategies, cloud provider, components deployment of virtual machine and its instances configuration. Component deployment to virtual machine instances involves the probability of making new components of already available components. Virtual machine instances configuration, relate to the instance type of virtual machine instances. CDOSim can model cloud deployments of software systems that were reverse engineered to KDM models. CDOSim has capability to represent the subscribers instead o the supplier's point of view. CDOSim is a simulator that permits the combination of fine grained models. CDOSim is best instance for runtime reconfiguration plans comparison or for finding the tradeoff between performance and costs [13].

J. XEN HYPERVISOR: -The Xen hypervisor is a layer of software operating directly on computer hardware substituting the operating system thus permitting the computer hardware to run numerous guest operating systems concurrently. It Supports a variety of platform and other general operating systems as guests operating on the hypervisor as provided [3, 4]. A computer operating the Xen hypervisor consists three components. It involves Xen hypervisor that operates directly on the hardware and becomes the interface for all hardware requests. By distinguishing the guests from the hardware, the Xen hypervisor is capable to run several operating systems independently and securely. The Domain 0 Guest known as Dom0 is established by the Xen hypervisor during at the time of system start-up and can operate any operating system except Windows. The Dom0 has unique privileges to access the Xen hypervisor that is not assigned to any other Domain Guests. The Domain Guests known as DomUs are established and managed by the Dom0 and independently run on the system. These guests are either run with a particular altered operating system known as Para-virtualization or un-modified operating systems leveraging special virtualization hardware (Intel VT and AMD-V) known as hardware virtual machine (HVM). The setup which utilizes in our situation is Para virtualization.

IV. DISCUSSION ON VISIONARY ASPECTS

In opposite to the FC and CC paradigms, DC depends on Information-oriented processing instead of Data oriented, which needs new, more elastic protocols and more productive programming models. In this way, (raw) data are context-free, while information is data going with meta-data. The meta-data positions the data in a particular context enabling more comprehensive knowledge discovery and analyses. The available computing paradigms, i.e. Fog and Cloud Computing, operate on large amounts of raw data created by particular Things, through predefined facilities.

However the raw data is out of context, the facilities required to be application-specific and tailored, needing data driven decisions. Creating an integrated scalable heterogeneous distributed computing environment from the level of Fog or Cloud is currently not plausible, as it does not allow the generic combination of all processing elements. On the other side, in the Dew Computing scenario, individual Things are responsible for gathering/producing the raw data, and are hence the only elements of the computing ecosystem which are totally known of the context the data were produced in.

As the Dew Computing is positioned at the lowest layer of the computing hierarchy, the context hence require to move from the bottom-level up utilizing well explained, information-oriented and standardized protocols of communication. Context-aware communication facilities have already been introduced before for application particular situations i.e. improved IP telephony [15]. We prefer that one of the possible directions for the growth of the Dew Computing could be by growth of an explored OSI model (Open Systems Interconnection), where a extra (eight) Context layer would be added on top of the available Application layer. The con- text information could then be passed via the left layers and automatically communicated by any network-enabled device, creating any device which utilizes this explored model a Dew-enabled device.

Since, not only communication protocols require to change. The primary programming principles of DC have to be of quite a different paradigm, as general programming models cannot cope effectively with the requirements of the Internet of Everything. Cloud Computing should be further explored to smoothly cope with the new existence of context in the data (the information), while still offering legacy for available facilities. For instance, Cloud Computing could then also be explored with a new layer which offers Context-as- a-Service (CaaS) on top of the available Software-as-a- Service (SaaS) layer.

On the CaaS layer, programming must be completely processing-oriented and transferring of information, not only the raw data. Novel programming models must be formulated which will permit interplay between CaaS services. Recently, concepts of context management systems and context graphs were introduced [11]which might introduce a possible direction in which the growth of the novel Dew-oriented programming models should be guided.

Another significant challenge that DC has to solve is: Personal High Productivity. To be capable to explain the requirements of high productivity in the way of the objective of CC/FC/DC, the expansion of the idea of High Productivity Computing, or, maybe better to say, High Productivity Information computing and processing has to be attempted. The idea of Productivity is changing with time to cover some facets and put less importance on other facets of the processes included. The aim of further growth is to extend the development of requirements in the field of high productivity scalable distributed computing. The developed visionary facet and strict definition of the idea of High Productivity in Information and Computing Sciences will be a significance attempt in the next computer science development phase.

A significant aspect of the Personal High Productivity Computing, which may be well used inside the DC principles, is equipment high productivity, or something probably best known service efficiency in the entire distributed computing hierarchy. Though the scientific usage of computing equipment was partially achieved from the Giga/Tera/Peta/Exa drive, the exploration was not mainly driven from the side of scientific requirements, but from the normal consumers. This can be noticed over the past some decades by scientific attempts put in to find other computer architectures and programming models, particularly adapted to dedicated requirements or generic kinds of programming principles.

As a development of two or more orders of magnitude of to-days computing systems is required in the future, involving systems with unprecedented amounts of different hardware, no of subscribers, and volumes of data, sustainability is critical to confirm the systems feasibility. Because of those requirements, currently there is a developing cross-domain interaction between the adoption of distributed programming paradigms and in Clouds high performance.

This type of mechanism will enable broad utilization of information processors to solve a myriad of ad-hoc problems by normal people, such as solving problems by not including particular computer system programming knowledge, but by enabling knowledge and context known “processors” and ”things” in the wider or immediate surrounding (environment) of the normal user (based on the request/algorithm) to offer, depending on the algorithmic environment, universal information and proper solutions.

On a practical level, the Dew Computing paradigm will be very significant in everyday living. So, for instance, a typical part of the Dew could be an joined traffic control system of a town, where each simple data processing/collection units positioned at all street segments among traffic lights would transfer information about traffic load in their region of responsibility, permitting a collective computerized overview of the complete situation, and enabling auto-adaptive traffic control nature.

CONCLUSIONS

Fog, Cloud and Dew distributed computing systems are the result of the exponential growth rate of computing and related techniques over the past five decades. This growth is the high driving force of the human society. It is required that novel computing techniques will regulate to emerge, i.e. today’s researched photonic, quantum computing paradigms and nano computing, that will regulate to make distributed computer systems more potential as compared to any standalone computer. This predicted growth, and the ever developing heterogeneity span tells that some of the present day beliefs of user-interaction, programmability and ad-hoc definition of user requirements will have to be heavily accommodated, defined and/or re-defined to enable the subscriber experience of a simple and integrated living environment.

The novel computing paradigm, Dew Computing, will concentrate on the three important points: Information processing (raw data and metadata explaining those data), High Productivity of subscriber-requests.

REFERENCE

[1] Nisha Peter, ”Fog Computing and Real time Application”, International Journal of Emerging

Technology and Advanced Engineering, Vol. 5, Issue 6, pp 266-269, June 2015.

[2] Kc gounda, Anurag Patro, Dines Dwivedi and Nagaraj Bhat "Virtualization approaches in cloud computing" International Journal of Computer Trends and Technology (IJCTT), Vol. 12, Issue 4, pp161-166, June 2014.

[3] Ivan Stojmenovic and Sheng Wen," The Fog Computing Paradigm: Scenarios and Security Issues", In the Proceedings of Federated Conference on Computer Science and Information Systems, Vol. 2, pp. 1–8, 2014.

[4] Swati Agarwal, Shashank Yadav and Arun Kumar Yadav," An Efficient Architecture and Algorithm for Resource Provisioning in Fog Computing", I.J. Information Engineering and Electronic Business, pp 48-61, January 2016

[5] Clinton Dsouza, Gail-Joon Ahn and Marthony Taguinod," Policy-Driven Security Management for Fog Computing: Preliminary Framework and A Case Study", IEEE IRI 2014, No. 13, pp 16-23, August 2014.

[6] K.P.Saharan and Anuj Kumar," Fog in Comparison to Cloud: A Survey", International Journal of Computer Applications (0975 – 8887), Volume 122 – No.3, pp 10-12, July 2015.

[7] T.Rajesh Kanna, M. Nagaraju and Ch.Vijay Bhaskar," Secure Fog Computing: Providing Data Security", International Journal of Research in Computer and Communication Technology, Vol 4, Issue 1, pp 53-55, January– 2015.

[8] D.C.Saste, P.V.Madhwai, N.B.Lokhande and V.N.Chothe," FOG COMPUTING: Comprehensive Approach for Avoiding Data Theft Attack Using Decoy Technology", International Journal Computer Technology and Application, Vol 5(5), pp 1768-1771, Sept.-Oct. 2014.

[9] Tom H. Luan, Longxiang Gao, Zhi Liz, Yang Xiang, Guiyi Wey, and Limin Sunz," Fog Computing: Focusing on Mobile Users at the Edge", arXiv:1502.01815v3 [cs.NI] , pp 1-11, 30 Mar 2016.

[10] Rajesh Bose, Murari Krishna Saha and Debabrata Sarddar," Fog Computing Made Easy with the Help of Citrix and Billboard Manager", International Journal of Computer Applications (0975 – 8887), Volume 121 – No.7, pp 19-23, July 2015.

[11] Viraj G. Mandlekar, VireshKumar Mahale, Sanket S.Sancheti and Maaz S. Rais," Survey on Fog Computing Mitigating Data Theft Attacks in Cloud", International Journal of Innovative Research in Computer Science & Technology, ISSN: 2347-5552, Volume-2, Issue-6, pp 13-16, November 2014.

[12] Durairaj. M and Kannan.P," A Study On Virtualization Techniques And Challenges In Cloud Computing", International Journal of Scientific & Technology Research, ISSN 2277-8616 , Volume 3, Issue 11, pp 147-151, November 2014

[13] Kamyab Khajehei," Role of virtualization in cloud computing", International Journal of Advance Research in Computer Science and Management Studies, ISSN: 2321-7782, Volume 2, Issue 4, pp 15-23, April 2014.

[14] Thogaricheti Ashwini and Mrs. Anuradha.S.G," Fog Computing to protect real and sensitivity information in Cloud", International Journal of Electronics and Computer Science Engineering, ISSN- 2277-1956, Volume 4, Number 1, pp 19-29

[15] Divya Shrungar J, Priya M P and Asha S M," Fog Computing: Security in Cloud Environment", International Journal of Advanced Research in Computer Science and Software Engineering, ISSN: 2277 128X, Volume 5, Issue 8, pp 803-807, August 2015.

[16] http://blogs.cisco.com/perspectives/iot-from-cloud-to-fog-computing

[17]http://www.webopedia.com/quick_ref/cloud_computing.asp

[18] Cisco, "Cisco delivers vision of fog computing to accelerate value from billions of connected devices," Cisco, Tech. Rep., Jan. 2014.

[19] Shanhe Yi, Cheng Li and Qun Li," A Survey of Fog Computing: Concepts, Applications and Issues", Mobidata'15, June 21, 2015.