IADIS International Journal on Computer Science and Information Systems
Vol. 10, No. 1, pp. 32-46
ISSN: 1646-3692

BIG DATA PROCESSING FOR SMART GRIDS

Driss Benhaddou*, Mohamed Riduan Abid#, Ouidad Achahbar#,
Nacer Khalil*, Tajjeeddine Rachidi# and Maen Al Assaf**

**University of Houston, Houston, TX, USA*
#Alakhawayn University in Ifrane, Ifrane, Morocco
***The University of Jordan, Amman, Jordan*

ABSTRACT

Smart Grids (SGs) are emerging as a promising technology meant to cope with the energy efficiency issue, currently witnessed in legacy electrical grids, by disseminating relevant information in a real-time mode among the different SG components.

The SG Advanced Metering Infrastructure (AMI) forms a central SG component, and consists basically of meters/sensors that are regularly communicating data towards the Control Plane. Much of these communicated data emanates from wireless sensors, and falls in the realm of Big Data. The latter needs substantial high-performance compute (HPC) power for processing and mining.

In this paper, we shed further light into a synergetic interface between SGs and the Cloud. We propose the use of Cloud computing to provide HPCaaS for SG Big Data processing, and delineate a suitable architecture. We present the blue print for deploying a real world private cloud testbed using OpenStack, Hadoop, and the MapReduce programming model. To assess the testbed functionalities, we run extensive experiments using benchmarked Big Data sets.

KEYWORDS

Smart grids, wireless sensors, cloud computing, high-performance compute

1. INTRODUCTION

Smart Grids are emerging as a promising technology to integrate renewable energy in the grid as well as customer site and cope with energy efficiency and thus addressing the worldwide stringent energy concerns (Satyajayant, Guoliang & Dejun 2012). In particular, energy efficiency is mainly fostered by the dissemination of information among producers and consumers of power in order to take appropriate decisions, mainly those relevant to the Demand/Response (DR) changes. Unlike traditional electrical grids where most of the components are automated and mostly exchange no data, SGs allow the exchange of real-time data for an efficient usage of the generated electrical energy through smart meters (Abid et al. 2013).

A major novelty in SG, when compared to ordinary electrical grid, is the two-way electricity flow: besides the ordinary electricity flow, which is from the producer (Electricity service provider) to the consumer, electricity flows on the other direction as well. Indeed, in SG, the consumer can produce electricity, mainly via the use of renewable energies (e.g., solar panels fixed on home rooftops) and inject it into the SG for sale. From the electricity service provider perspective, this stipulates a concise metering of produced electricity and the consumed one, i.e., Demand/Response (DR). To meet a concise tracking of the DR variances, data needs to be communicated among concerned SG components and processed in a real-time manner.

On the other hand, microgrids have been developed as a mean to integrate Distributed Energy Resources (DER), such as photovoltaic (PV), micro-turbines (MT) and fuel cells (FC), directly at the customer site (Wissner 2011). Microgrids provide reliable power with economic, environmental and technical benefits. More specifically, microgrids are currently being developed as a potentially effective strategy to feed power directly to low voltage (LV) networks, thereby allowing the customer to become an active participant in the grid (Lopes et al. 2006). In certain instances, retail electricity providers (REP) developed by communities (i.e. community microgrids) share an interest in such microgrids and interact with nearby microgrids within the smart grid to deliver electricity to consumers (see Fig 1).

Indeed, SG is moving toward an architecture of interconnected microgrids (see Fig. 1), where microgrids composed of buildings (e.g., residential, business, or industrial), where every electrical appliance and DER is equipped with wired and wireless sensors and actuators that sense electricity consumptions and productions and receive commands for control operations (e.g., switching On/Off and adjusting the consumption behaviors). This is further eased by the recent advances in the Electronics industry which promoted cheap manufacturing of these sensors. Taking into account the gigantesque number of needed sensors, and the frequency by which data metering occurs (e.g., 1 Hz), the produced data falls in the scope of Big Data as it exhibits the three basic characteristics of Big Data, i.e., Volume, Variety, and Velocity (aka., The Big Data three Vs). The processing of such real-time data still presents challenges merely because the generated data falls in the realm of Big Data.

To process Big Data, substantial high performance compute power (HPC) is needed. Cloud computing provides different kinds of services, e.g., HPCaaS (High performance Computing as a Service) which is the means for providing HPC, e.g., for Big data processing. With Cloud services, the end-user (e.g., SG operator) is provided with a Cloud application interface via which he can input his Big Data, request specific processing, and get relevant input. To provide cloud services, different cloud deployments models arise, basically public, private and community clouds.

In this paper, we propose the use of the private cloud computing model to provide HPCaaS for SG operator. We delineate a synergetic interfacing between the wireless sensors, SG, and the Cloud. Along, we present an architecture for AMI deployment using wireless sensors. The latter are used to control electrical appliances control in smart buildings. A blueprint for deploying a real-world private cloud for HPCaaS is presented. This uses OpenStack along with the Hadoop's MapReduce programming model.

The rest of the paper is organized as follows: Section II covers key elements of WSNs as a key technology enabler for the Advanced Meeting Infrastructure. In Section III, we highlight AMI (Advanced Metering Infrastructure) and present a relevant architecture for deployment in

Residential buildings. Section IV gives detailed description of our proposed architecture in terms of integrating SG-WSN on one hand, and SG-Could on the other. Section V outlines the main deployment steps of the HPCaaS platform. Experimentations are presented in Section VI. Finally, we conclude and present future work in Section VII.

2. COMMUNITY MICROGRIDS AND ENERGY INFORMATION SYSTEMS

Figure 1 shows a typical architecture of a microgrid. It is composed Microgrid Central Controller (MGCC) that interacts with local microgrid controllers (MC) to implement energy management functions. MCs are typically interfaced with DER, such as Photovoltaic (PV), at rooftops of smart buildings for instance, and implement local control of these resources. These components control electricity flow in the microgrid as well as communicate information through Energy Information System (EIS). EIS plays a key role in managing the resources within the microgrid and can be thought of as a layer on the top of the power layer (see Figure 2). EIS has the objective of making sure microgrid is stable, reliable, and resilient (can work in normal or islanded mode). EIS has also the capability of interacting with the smart grid market as well as other nearby microgrids.

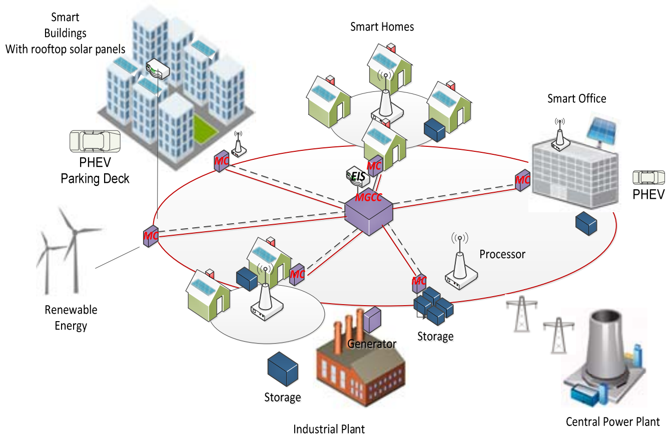

Figure 1. General components and architecture of a Microgrid.

We model EIS of a community microgrids as a group of functions that represent different stake holders in the microgrids (see Fig. 2) namely users/occupants, DER are local produce of energy, buildings that include consumers such as machines and computers, etc. These components produce the data that is used to develop energy prediction model, user preference and activity models, consumption profile models. The data is collected by our middleware and archived by EIS for future use and further development. The data is accessed by a number of applications including, solar energy prediction, user activity modeling, consumption prediction, thermal and air flow setting, HVAC control, event generation for maintenance and failure detection, etc.

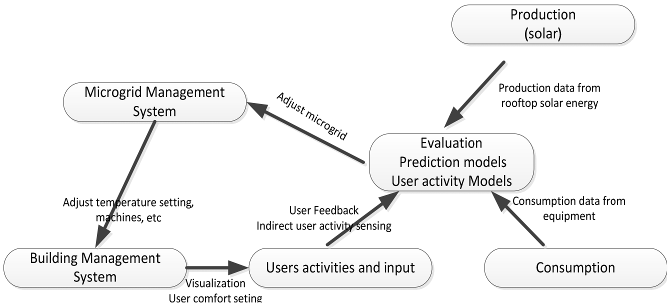

Figure 2. Architecture of the EIS system.

The events associated with this model are collected through a wireless and wireline network and archived in the cloud. Networking as well as computing time will therefore play a key role in the delay of the feedback process and need to be quantified, which is the focus of the performance evaluation in this paper.

3. SMART GRID AND CLOUD COMPUTING

Smart gird related research and development is being investigated by utility companies, standards bodies, and university research groups, and is currently one of the top technologies that will give the economy a competitive advantage. Several consortiums have been established to develop technologies that will enable the migration of the current electric power grid toward a reliable and efficient smart grid (Bhatnagar & Rao, 2005, EIA team 2013, Zyga 2011, Johnson Controls's team 2013). The US Department of Energy has initiated a Modern Grid Initiative (MGI) to investigate the key technologies that are needed to enable smart grid (Abid et al. 2013). The demand for microgrids as a strategy for integrating small scale DERs, ensuring power grid reliability, energy independence and efficiency is growing. Studies are being conducted worldwide to assess various topics including DERs integration, smart microgrid controllers, network based control and management, demand side management, stability, and renewable energy integration. The energy generated by DERs are variable and therefore microgrids become a source of non-controllable power. The impact of these DERs on Low Voltage (LV) network in terms of power balance (Strbac 2008), voltage rise (Masters 2002), quality and stability (Löf, 2011) become significant as the number of customer installing DERs in their site increase. These instabilities may propagate to the power grid operation as it becomes challenging to manage a distribution system with a large amount of non-controllable sources that inject reverse variable power. The FREEDM systems aim to develop innovative technologies, scalable and secure communications, and distributed control (Berkat 2011). The Cyber Physical Challenges of Transient Stability and Security in Power Grids project (Lopes et al. 2006) creates a cyber-physical system capable of adjusting the loads, communicating the information between different parties and sense abnormal states caused by natural faults or malicious attacks. As a way to address the challenge of matching demand to supply, projects are addressing issues such as robust network connectivity, resilient electric power infrastructure, robust control (Hairong et al. 2011, Tsado et al. 2014, Sanders 2012, Massoud 2014), modeling and predicting human behavior and activity in building usage pattern (Dong et al. 2011, Philipose et al. 2004, Nugent & Augusta 2006, Hong et al. 2009, Yu , Bruckner & Velik, 2010, Stephenson et al. 2010).

Development of DER controllers for efficient and effective operation of microgrids with control strategies using Information and Communication Technologies (ICT) has been investigated (Gopalan et al. 2012, Ganet al. 2013, *More Microgrids*). The findings in these studies underscore the importance of ICT technologies and recommend further research in exploiting ICT for smart grid application. ICT virtualization through Software Defined Networks (SDN), Network Functions Virtualization (NFV) and Cloud computing to accomplish smart grid objectives and enable real time communication and control in smart grid has attracted the attention of researchers in smart grid. In particular, cloud computing will provide a platform that can implement the following smart grid functions (Bera et al. 2015):

- Energy Management: This is one of the key function in smart grid and include demand side management (demand response), Building/Home energy management systems, and DER optimization.
- Support of multiple and heterogeneous devices.
- Information management and integration through scalable and unified way of information representation and integration.
- Support of layered and heterogeneous architecture to help implement a complex system such as smart grid.
- Implement security measured from data and information perspectives.

Several of the end systems generating data are connected to the smart grid through wireless sensors, the following section will talk about the Wireless sensors data communications.

4. WIRELESS SENSORS DATA COMMUNICATION IN SGS

Thanks to their ease-of-deployment and self-healing features, WSNs (Wireless Sensor Networks) are becoming essential to SG (Akyildiz et al. 2002). They enable the real-time tracking of electrical appliances' consumption levels, and ease the control of these appliances by turning them On or Off remotely.

SG relies on AMI (Advanced Metering Infrastructure) for introducing a two-way power and information communication between different producers and consumers in the grid (Luhua et al. 2010). Thanks to this technology, Distributed Energy Resources (DER) became a pillar in SG, but also came with challenges, e.g., the fact that Utility Companies can no longer control the production entirely and this might lead to grid instability if it is not well managed (Rugthaicharoencheep et al. 2012). However, thanks to AMI and WSNs, the utility may be able to not only control production but also consumption through Demand Response (Datchanamoorthy et al. 2011).

Figure 3 shows a deployed network diagram for a WSN in a residential building (Khalil et al. 2014). The WSN uses Zigbee (*The ZigBee Alliance)* at the access layer and IPv6 as network protocol. The WSN sink is connected to the gateway server via a USB link. The latter connection serves as a tunnel between the WSN and the gateway server. The gateway server is connected to both the WSN and the outside network using Wi-Fi (at the access layer) and IPv4 (at the network layer). The main role of the gateway server is to promote communication between these two networks that are not based on the same network technologies. In fact, a process running in this server keeps track of the packets flowing and transmutes them to fit both networks' packet formats.

The gateway server is connected to a middle-ware server that masks the heterogeneity of the data. It receives the data from the WSN, filters it, transforms it, and stores it. This data is then used by the EMS to make decisions on how to control the building for higher efficiency; it is also used to give end-users access to real-time consumption information.

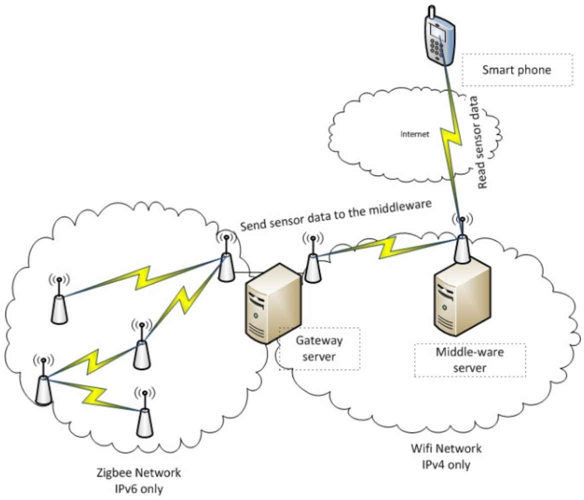

Figure 3. Network Diagram of the WSN-enabled building

5. ADVANCED METERING INFRASTRUCTURE (AMI) AND RENEWABLE ENERGY INTEGRATION

The Advanced Metering Infrastructure (AMI) is the SG component that manifests the "smartness" in SGs. This is a networking infrastructure that connects SG components and allows for real-time dissemination of data. The latter can be either raw data to be processed or control commands to carry out specific tasks, e.g., monitoring the operation of SG components.

In a former relevant work (Abid et al. 2013), an AMI architecture, that connects smart buildings into SG, has been presented. This consists on the use of wireless mesh networks (Akyildiz & Wang 2005) as the underlying networking technology as explicated in Figure 4.

This architecture deems the SG as a set of micro-grids that are independent of each other, and exhibit the main faculty of being able to connect and disconnect to the main SG depending on generated electricity and predicted demand. This interconnects four main Components:

1. *Residential Smart Micro-Grid*: This consists of residences, along with the corresponding electrical appliances. The appliances are equipped with Zigbee sensors and form a wireless mesh network.
2. *Control Plane:* This is the back office where all computation and data processing are done.

3. *End user (Home Owner):* capable of tracking the energy consumption at his residence, and interacting with the Application Server at the Control Plane..
4. *Electricity Provider:* has access to the real-time data collected at the Control Plane.

The backbone network can be a private one (e.g., operated by the electricity provider) or a public (e.g., the Internet). This depends mainly on the degrees of privacy and security to maintain.

This architecture provide a flexibility for a microgrid (e.g residential) to be independently managed in connected and isolated mode. The operation of a microgrid is influenced by other interconnected microgrids and the “big” Untility grid. Indeed, this proliferation of microgrids will influence the future architecture of the smart grid which will be composed of interconnected microgrid similar to how the Internet is a mesh of interconnected networks. Cloud computing is revolutionizing the way services are being implemented and offered in Internet and will do the similar impact in the smart grid. Cloud computing has the ability to implement many services as a platform in smart grid and will virtualize their function, allowing for smart grid function virtualization (SGFV).

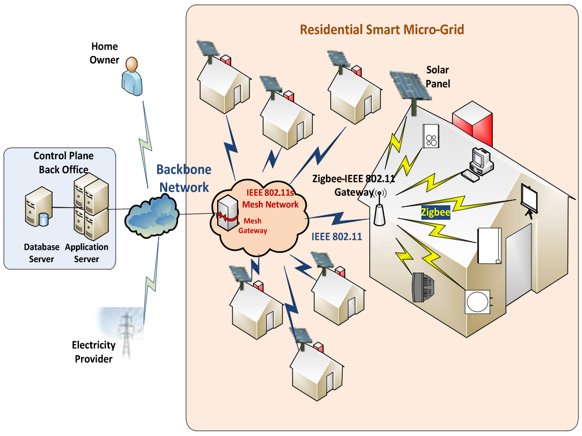

Figure 4. WSN-based Smart Grid Advanced Metering Infrastructure (AMI) Architecture for Smart Buildings (Abid et al. 2013)

6. SMARTS GRIDS AND THE CLOUD: THE INTERFACE

There is an inherent and synergetic “matching” between SGs and Cloud computing: on one hand, SGs are generating Big Data and are in crucial need of storage and processing power. On the other hand, Cloud computing has been tailored with the main goal of providing compute power (e.g., storage and processing) as a utility. A fact that further eases this “matching” is that Cloud computing masks all the burden of deploying and maintaining the needed IT infrastructure, e.g., recruiting engineers, maintenance, hardware and software purchase, etc. In addition cloud computing provide benefit to meet scalability where elastic computing resources are provisioned on-demand according to the actual computing needs.

In this paper, we propose the use of an HPCaaS (HPC as a Service) Cloud Computing Platform SG Big Data processing. Our proposed HPCaaS uses Hadoop (Apache Hadoop) as the clustering platform to store and process data. This uses HDFS (Hadoop Distributed File System) for data storage and the MapReduce programming model (Dean & Ghemawat 2004) for distributed data processing, and both run on top of a Cloud computing platform: the Openstack platform (*Open source software for building private and public clouds*). The Big Data generated by SGs (mainly generated by the SG sensors and meters) falls in the category of K-V (Key-value) pairs (e.g., sensor id, timestamp, consumption levels, etc).

The Hadoop HDFS manages storage and related issues (e.g., chunks replication, failure recovery), and the Hadoop MapReduce runs the relevant "jobs" on selected chunks of the big data. The proposed architecture is depicted in Figure 5.

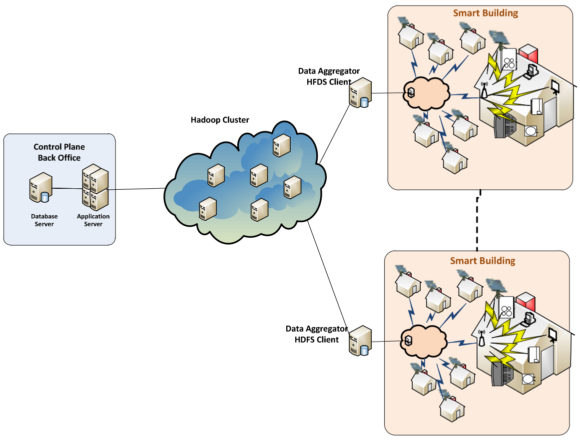

Figure 5. Smart Grid and Cloud interface general architecture

The main idea behind this architecture is to have an appliance server (Data aggregator) attached to each smart microgrid (i.e., a set of buildings located in the same geographical area). This device has two interfaces: 1. Data aggregator which collects data from sensors and meters, and 2. HDFS client which interacts with HDFS, and forwards data, see Fig. 6.

The data aggregator process receives Key-value pairs from the different buildings and forward them to the HDFS client process. The latter communicates with the data aggregator process via an IPC (Inter Process Communication) protocol, and establishes a connection with HDFS residing in the Cloud.

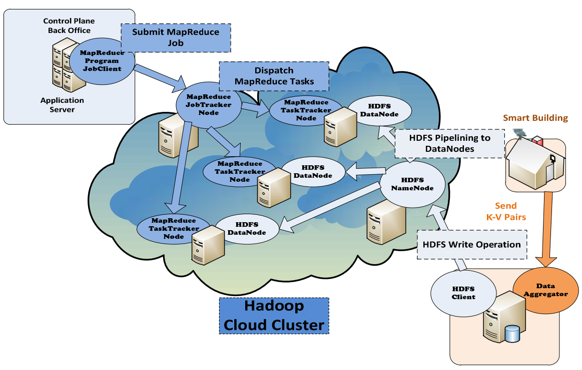

Figure 6. Data Flow Architecture

HDFS has two types of nodes: 1. Namenode (the master), which manages the file system by keeping relevant files metadata and namespace entries, and 2. Datanodes (workers), which are the real workhorses of the system. They store and retrieve files' blocks upon request from the master node (i.e., the namenode). The communications are done via the RPC (Remote Procedure Call) protocol. This way, the received K-V pairs will be stored by the HDFS while automatically maintaining replication (replication factor is set to 3, by default) and coping with node failure.

On the other direction, the application server issues MapReduce jobs to run on the big data chunks stored by the HDFS. This involves four main entities:

1. *The MapReduce Jobclient:* resides in the Application server and is responsible of establishing the connection with the Hadoop cluster and submitting the MapReduce Jobs.
2. *The JobTracker:* coordinates job execution by splitting the main job into tasks, and delegates them to other nodes (the TaskTrackers) while accounting for two main factors: load balancing and location of the files in the HDFS namenodes.
3. *TaskTracker:* The MapReduce horseworkers that run the tasks is assigned by the jobtaracker.
4. *HDFS:* is responsible for providing and sharing the files with the tasktrackers.

Indeed, HDFS stores data and makes it available for requests and jobs emanating from the application server. After running relevant processing (e.g., by executing relevant Hadoop MapReduce jobs), the result is forwarded back to the control plane.

Ideally, the end-user interface will be deployed at the application server. This will allow the end-user (e.g., SG monitor) to run specific queries against the data residing in the Hadoop cluster. On the other side, the data aggregator will continuously run `"processes" asking the Hadoop cluster to store generated data.

7. HPCAAS PRIVATE CLOUD DEPLOYMENT

In this section we delineate a blueprint for deploying a real-world HPCaaS private cloud deployment. We used the OpenStack open-source Cloud Computing platform (*Open source software for building private and public clouds*), and Hadoop parallel and distributed open-source system (Apache Hadoop).

7.1 OpenStack Deployment

HPCaaS private cloud deployment starts with installing *OpenStack*. Because we have used small experimental setups (in terms of storage and processing), it was sufficient to deploy the following *OpenStack* components: *Keystone, Glance, Nova* and *Horizon*. These components can provide both data storage and data processing to implement *HPCaaS*.

After installing and configuring the KVM Virtualization Hypervisor (*The Kernel Based Virtual Machine*), the first OpenStack component that was installed is the Keystone. This component manages Authentication, e.g., by creating relevant tenants (OpenStack projects), associated users, and roles. The second OpenStack component, we installed, is the Glance. This creates and manages the different formats of virtual machines images. Glance package includes glance-api that accepts incoming API requests; glance-database that stores all information about images, and finally glance-registry that is responsible of retrieving and storing metadata about images. The third component is the Nova package. This contains nova-compute, nova-scheduler, nova-network, nova-objectstore, nova-api, rabbitmq-server, novnc and nova-consoleauth. All these components collaborate and communicate with each other to create and manage virtual machines (VMs) (Pepple 2011).

7.2 Hadoop Deployment

Hadoop deployment starts with identifying the master and slave nodes. For master node, there are six files that need to be configured: *core-site, hadoop-env, hdfs, mapred-site*, master and slaves files. Concerning slave nodes, the only files that need to be configured are *hadoop-env, core-site, hdfs* and *mapred-site* files. These files aim at setting environment variables, defining common properties (e.g. HDFS and MapReduce properties), specifying the master and slave nodes, setting the number of replicas, etc.

After configuring all needed files, nodes have to communicate with each other via the SSH protocol (Secure Shell). Next, HDFS namenode was formatted. This cleans the filesystem and creates storage directories. Finally, the Hadoop cluster can be launched to run jobs after starting the HDFS and MapReduce daemons. Detailed Hadoop documentation is provided by Noll guidelines.

8. EXPERIMENTATION

These experimentations are meant for proof of concept demonstration as well studying the cloud resources needed for a typical microgrid/smar grid function and to what extend the elastic feature of cloud can be exploited. As the amount of data increase, more compute power is needed. However, in certain instance, this compute power would need to be reduced as the amount of data decrease, thus taking advantage of the elastic feature of the cloud. Our vision is to implement the Energy Information System model described in section 2 in using the cloud. Abid et al. (2013) showed the importance of WSNs and middleware design; Khalil et al. (2014) deployed a real-word WSN, in a residential area, for energy management; in this paper, we show the importance of Cloud services and the necessity of adding Cloud related APIs into the middleware. For the purpose of demonstrating the concept, we used a single 8-core server (Dell PowerEdge with 6GB of RAM) in which we forked 8 VMs, set them as a HPC Cluster, and run relevant experiments. The installed software flavors are Hadoop version 1.2.1 and OpenStack (Nova, Keystone, Glance and Horizon) Folsom release.

The results showed tangible stability with moderate data sizes. However, when the data size grows, the virtual cluster could not afford the needed compute power. Thus, we plan to deploy a much powerful cluster using multiple servers instead of a single one, especially that the concept and the functionalities have been assessed in this work. We will also develop tools that keep track of different delays in the network and develop a structure of the networks. This structure will give an indication when to grow and shrink resources taking advantage of the elasticity model in cloud computing.

We run several experiments for writing, reading, and sorting benchmarked Big Data. The latter is similar in structure to the one generated by SGs. The tested Big Data follows a Key-Pair structure, and consists basically of random data formatted as follows: 10 bytes for key, 10 bytes for row identifier, and 78 bytes for filler (letters from A to Z) (Noll).

8.1 Data Sets

We used TeraSort and TestDFSIO (Tests for Distributed File System I/O) (Noll). These are well-known benchmarks. TeraSort was developed by Owen O'Malley and Arun Murthy, at Yahoo Inc. It won the annual general purpose terabyte sort benchmark in 2008 and 2009. It does considerable computation, networking, and storage I/O. TeraSort performance metrics consist of measuring the average time to sort a given datasets.

TestDFSIO benchmark is used to check the I/O rate of the Hadoop cluster with write and read operations. Such benchmark is helpful for testing HDFS by checking network performance, and testing hardware, OS, and Hadoop setup. TestDFSIO performance metrics consist of measuring the execution time to write (TestDFSIO-Write) and read (TestDFSIO-Read) datasets.

For TeraSort, we used 100 MB, 1 GB, 10 GB and 30 GB datasets; for TestDFSIO, we used 100 MB, 1 GB, 10 GB and 100 GB datasets.

We started experimentation by gradually scaling up the cluster granularity from 3 to 8t VMs. We started by 3 as this is the default Replica factor in HDFS; For each benchmark, we run three tests for each dataset size, and calculated the mean to avoid any outliers.

8.2 Results and Analysis

The results of running TestDFSIO and TeraSort on the cluster's VM instances are illustrated in Figures 7-9.

For all dataset sizes, the overall performance for running TestDFSIO-Write (see Fig. 7) is quasi stable as the number of VMs increases from 3 to, 4 and 5. TestDFSIO-Read exhibited quite the same performance in terms of stability when the VM machines granularity is less or equal to 5.

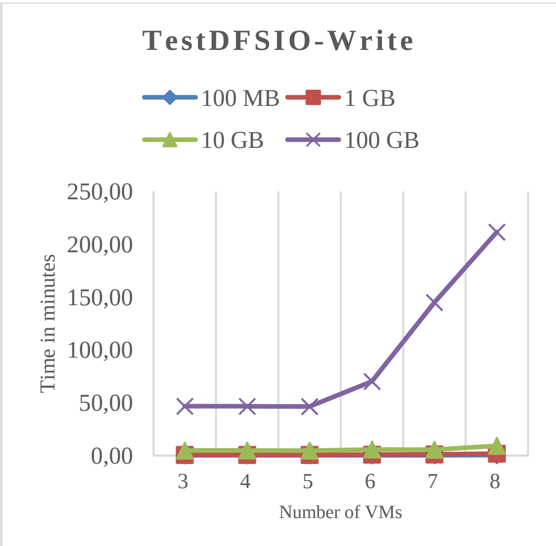

Figure 7. TestDFSIO-Write performance

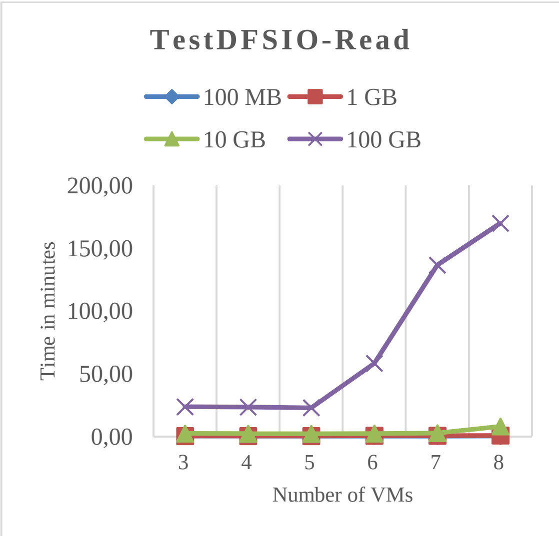

Figure 8. TestDFSIO-Read performance

Indeed, in Figure 8, we see that reading the different dataset sizes keeps the same performance as the number of VMs increases from 3 to 5. However, when scaling up the cluster granularity to 6-8 VMs, the performance of both TestDFSIO write and read operations decreases for all dataset sizes. We explain this fact by the scarcity of compute power in terms of available memory and processing power; especially, that frequent context switches will occur as the hypervisor has to frequently switch from a VM instance to another; besides, the server has to cope with 3 replicas of the datasets, each is of 100 GB, and since the memory (6 GB) cannot afford this, the hypervisor has to keep swapping in/out memory images from/to virtual memory: a process which is time and resource consuming.

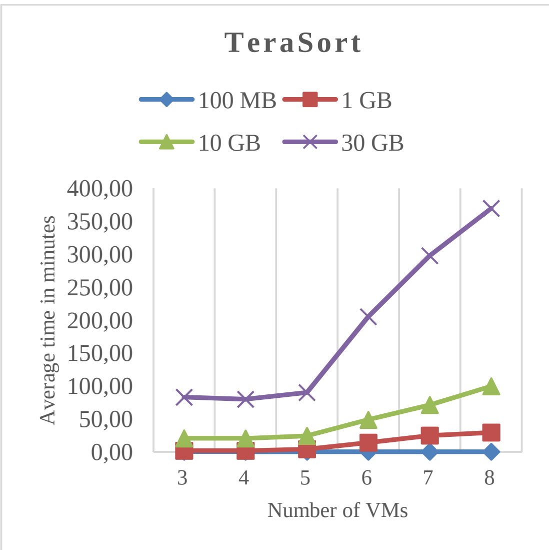

Figure 9. TeraSort performance

The same applies for the TeraSort operation (see Fig. 9). This shows also a stable performance when running TeraSort benchmark on 3-5 VMs, and a sharp decrease when scaling up to 6-8 VMs, for bigger data sets, e.g., 10 GB and 30 GB.

9. CONCLUSION

As formerly stated, the aim from these experiments is assessing the operational functionalities of the testbed. A testbed with more powerful compute power will definitely cope with larger data sets and more VMs; this constitutes the main step ahead in our future work.

As future work, we intend to (1) scale up the hardware used in the testbed in order to further assess, through experimentation, the HPCaaS performance needed for processing SG data in real-time; (2) elaborate a middleware architecture with specific APIs for gluing together the various heterogeneous components of the SG on one hand, and the Cloud services on the other; (3) evaluate the structure of the cloud in term of delay and throughput to see what smart grid functions could be implemented in the cloud and what functions could be implemented locally near the resource. Indeed local versus global processing selection is still under investigation by researchers.

REFERENCES

A. M. Löf, 2012. Testing of low voltage network automation. Master of Science Thesis. Tampere University of Technology.

A. Berkat, 2011. Using Case-Based Reasoning (CBR) for detecting computer virus. J. Comput. Sci., vol. 8, no. 4, pp. 606–610.

A. Massoud, 2014. A Smart Self-Healing Grid: In Pursuit of a More Reliable and Resilient System [In My View]. Power and Energy Magazine, IEEE , vol.12, no.1, pp.112,110.

Apache Hadoop: http://hadoop.apache.org/

B. Dong, K. P. Lam, C. P. Neuman, U. Technologies, and E. Hartford, 2011. Integrated Building Control Based On Occupant Behavior Pattern Detection And Local Weather Forecasting. vol. 3, pp. 14–16.

C. L. Masters, 2002. Voltage rise - the big issue when connecting embedded generation to long 11 kV overhead lines. Power Engineering Journal, vol. 16, pp. 5-12.

C. Nugent and J. Augusta, 2006. A logical approach to ADL recognition for alzheimer’s patients. Smart Homes Beyond ICOST 4th Int. Conf. Smart Homes Heal. Telemat., vol. 1, no. 819, pp. 1–8.

D. Bruckner and R. Velik, 2010. Behavior Learning in Dwelling Environments With Hidden Markov Models. Ind. Electron. IEEE Trans., vol. 57, no. 11, pp. 3653–3660.

EIA team, 2013. Heating and cooling no longer majority of U.S. home energy use. US. Energy Information Administration. Online: http://www.eia.gov/todayinenergy/detail.cfm?id=10271#.

F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, 2002. Wireless sensor networks: a survey. The International Journal of Computer and Telecommunications Networking, Vol. 38, No. 4, pp. 393-422

F. Akyildiz, X. Wang, 2005. Wireless Mesh Networks: A Survey. Computer Networks and ISDN Systems, Vol. 47, No. 4, pp. 445-487.

Gopalan, S.A.; Sreeram, V.; Iu, H.H.C.; Xu, Z.; Dong, Z.Y.; Wong, K.P., 2012. Fault analysis of an islanded Multi-microgrid. Power and Energy Society General Meeting, IEEE, vol., no., pp.1,6, 22-26.

Gan, L.K.; Macpherson, D.E.; Shek, J.K.H., 2013. Synchronisation control and operation of microgrids for rural/island applications. Power Engineering Conference (UPEC), 48th International Universities, vol., no., pp.1,6, 2-5.

G. Strbac, 2008. Demand side management: Benefits and challenges. Energy Policy, vol. 36, no. 12, pp. 4419-4426.

Hairong Qi; Xiaorui Wang; Tolbert, L.M.; Fangxing Li; Peng, F.Z.; Peng Ning; Amin, M., 2011. A Resilient Real-Time System Design for a Secure and Reconfigurable Power Grid. Smart Grid, IEEE Transactions, vol.2, no.4, pp.770,781.

J. Dean, S. Ghemawat, 2004. Mapreduce: Simplified data processing on large clusters. In Proceedings of the 6th USENIX OSDI, pages 137–150.

J. A. P. Lopes, C. L. Moreira, and A. G. Madureira, 2006. Defining control strategies for microgrids islanded operation. IEEE Transactions on Power Systems, vol. 21, pp. 916-924.

Johnson Controls’s team, 2013. CENTRAL PLANT OPTIMIZATIONTM. Online: http://www.johnsoncontrols.com/content/us/en/products/building_efficiency/productsand-systems/integrated_hvac_systems/Central_Plant_Optimization.html.

J. Stephenson, B. Barton, G. Carrington, D. Gnoth, R. Lawson, and P. Thorsnes, 2010. Energy cultures: A framework for understanding energy behaviours. Energy Policy, vol. 38, no. 10, pp. 6120–6129.

KVM. The Kernel Based Virtual Machine. Online: http://www.linux-kvm.org/page/Main_Page

K. Pepple, 2011. Deploying Openstack. O'Reilly Media.

L. Zyga, 2011. US energy use chart shows we waste more than half of our energy. Online: http://phys.org/news/2011-04-energy_1.html.

M. Satyajayant, X. Guoliang, Y. Dejun, 2012. Smart Grid – The New and Improved Power Grid: A Survey. IEEE Communications Surveys & Tutorials, Vol. 14, No. 4, Fourth Quarter.

M. R. Abid, A. Khallaayoun, H. Harroud, R. Lghoul, M. Boulmalf, D. Benhaddou, 2013. A Wireless Mesh Architecture for the Advanced Metering Infrastructure in Residential Smart Grids. IEEE GreenTech.

More Microgrids. Online: http://www.microgrids.eu/default.php (accessed: 14, December, 2014).

M. R. Abid, A. Khallaayoun, H. Harroud, R. Lghoul, M. Boulmalf, and D. Benhaddou, 2013. A Wireless Mesh Architecture for the Advanced Metering Infrastructure in Residential Smart Grids. IEEE Green Technol. Conf., pp. 338–344, Apr. 2013.

M.G. Noll. Running Hadoop on Ubuntu Linux (Multi-Node Cluster). Online: http://www.michael-noll.com/tutorials/running-hadoop-on-ubuntu-linux-multi-node-cluster/

M. G. Noll. Benchmarking and Stress Testing on Hadoop Cluster with TeraSort, TestDFSIO & Co. Online:http://www.michael-noll.com/blog/2011/04/09/benchmarking-and-stress-testing-an-hadoop-cluster-with-terasort-testdfsio-nnbench-mrbench/

Matthias Wissner, 2011. The Smart Grid – A successful of secrets. Applied Energy, 88 2509–2518.

M. Philipose, K. P. Fishkin, M. Perkowitz, D. J. Patterson, D. Fox, H. Kautz, and D. Hahnel, 2004. Inferring activities from interactions with objects. Pervasive Comput. IEEE, vol. 3, no. 4, pp. 50–57.

N. Rugthaicharoencheep, M. Boonthienthong, 2012. Smart grid for energy management on distribution system with distributed generation. IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER).

N. Khalil, M. R. Abid, D. Benhaddou, M. Gerndt, 2014. Wireless Sensor Networks for IoT (Internet of Things). Proceedings of the IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP). Singapore.

R. Bhatnagar and C. Rao, 2005. Energy Resource Management Based on Data Mining and Artificial Intelligence Current Research Related Data Mining Research. pp. 14–23.

Sanders, W.H., 2012. Building resilient infrastructures for smart energy systems (abstract). Innovative Smart Grid Technologies (ISGT), IEEE PES , vol., no., pp.1,1, 16-20.

S. Bera, S. Misra, Joel J.P.C. Rodrigues, 2015. Cloud Computing Applications for Smart Grid: A Survey. IEEE Trans. On Parallel and Distributed Systems, Vol. 26, No. 5.

S. Datchanamoorthy, S. Kumar, Y. Ozturk, G. Lee, 2011. Optimal time-of-use pricing for residential load control. EEE International Conference on Smart Grid Communications (SmartGridComm).

Tsado, Y.; Lund, D.; Gamage, K., 2014. Resilient wireless communication networking for Smart grid BAN. Energy Conference (ENERGYCON), IEEE International, vol., no., pp.846,851, 13-16.

The ZigBee Alliance, online: “http://www.zigbee.org/”

The Openstack Cloud Software. Open source software for building private and public clouds. Online: http://www.openstack.org/

T. Yu, 2010. Modeling Occupancy Behavior for Energy Efficiency and Occupants Comfort Management in Intelligent Building. Ninth International Conference on Machine Learning and Applications, pp. 726–731.

X. Hong, C. Nugent, M. Mulvenna, S. McClean, B. Scotney, and S. Devlin, 2009. Evidential fusion of sensor data for activity recognition in smart homes. Pervasive Mob. Comput., vol. 5, no. 3, pp. 236–252.

Z. Luhua, Y. Zhonglin, W. Sitong, Y. Ruiming, Z. Hui, Y. Qingduo, 2010. Effects of Advanced Metering Infrastructure (AMI) on relations of Power Supply and Application in smart grid. China International Conference on Electricity Distribution (CICED).