

# On Edge Cloud Service Provision with Distributed Home Servers

Amin M. Khan<sup>\*‡</sup>, Felix Freitag<sup>†</sup>

<sup>\*</sup> Department of Computer Science, UiT The Arctic University of Norway. Tromso, Norway  
amin.khan@uit.no

<sup>†</sup> Department of Computer Architecture, Universitat Politècnica de Catalunya. Barcelona, Spain  
felix@ac.upc.edu

<sup>‡</sup> Hitachi Vantara Corporation. Lisbon, Portugal  
amin.khan@hitachivantara.com

**Abstract**—Edge computing has been proposed for new types of cloud services, which need computing infrastructure at the network edge. Driven by important use cases from the Internet of Things (IoT) domain, edge cloud computing has also a huge business potential. Edge computing devices are already operational in many industrial and consumer-oriented scenarios. A typical characteristic of these solutions is, however, that the hardware and software platforms in use are proprietary and closed. The interaction among the different vendor platforms, and their extension with third party services, in general, is not well supported. As a consequence, the opportunities for third party providers to build new tailored edge services on these platforms face significant barriers. In this paper we argue to build a collaborative edge cloud deployed on home servers. We position the proposed system and describe its building elements, where we use Docker containers for provisioning the services.

**Index Terms**—edge cloud computing; community clouds

## I. INTRODUCTION

Edge computing combines the advantages of data center cloud computing, such as resource and service elasticity, with the capacities of local processing. From a traditional cloud service provider perspective, edge computing infrastructure is mainly a resource that extends the capabilities of a data center-based cloud service to the network edge.

Edge computing solutions are already operational in many industrial and consumer-oriented scenarios, covering major application domains. Important vendors nowadays provide edge computing solutions, e.g. [1], [2]. While we can observe a tendency of large industrial players to contribute to open platforms, such as Eclipse [3], many of the solutions currently on the market are based on closed hardware and software platforms, which limit service extensions and the deployment of framework-independent third-party services.

In this paper we argue for an extensible collaborative edge cloud, which is formed by user-provided computing and communication resources to allow providing services at the network edge. The software platform, which is installed on these devices, must be open for extension with additional services by the participants. In the collaborative edge cloud computing model, the main cloud resources are the spare resources of distributed edge devices. This way the elasticity

in resources and service provision at the network edge can be obtained. Such an edge cloud, however, will also leverage data center cloud services to improve service performance, e.g. for higher resilience.

In the rest of the paper, we present our case for an open platform at the edge in Section II, and discuss related work in Section III. We propose the architecture in Section IV, and present its prototype implementation in Section V. We conclude our findings in Section VI.

## II. THE CASE FOR AN OPEN PLATFORM AT THE EDGE

We see the evolution of the Internet of Things (IoT) domain as an important driver for the proposed system, since it has created many use cases that target working with edge clouds [4]. These use cases will require tailoring the service solutions to specific customer needs, for which an existing vendor framework is adapted or extended. Such software solutions are, therefore, specific to a particular vendor's platform, and so are not easily portable to operate with the other vendors' frameworks. Small and medium enterprises (SMEs), which often implement such customized solutions, would benefit if vendors' edge platforms are more flexible in sharing services among edge devices and if the platforms offer the deployment of third-party services.

The pooling of resources at the network edge through a set of common basic services could lead to edge clouds, where nodes collaborate in provisioning the different services. This collaboration consists of the participants who accept to run some support services, such as those for interconnection and management of devices, which are needed to enable the collaborative edge cloud. The administration of this cloud is decentralized, however, consisting of multiple administrative domains, which correspond to the independent owners of the devices.

For commercial service providers, a collaborative edge cloud should provide: 1) a reduced entry barrier, taking advantage of the availability of shared resources, 2) the knowledge about the infrastructure, which needs to be transparent and neutral, i.e. there are no barriers that artificially limit the scope of services

that can be created, and 3) the access for service providers to deploy applications in privately owned edge devices.

From the perspective of individual users, the operational model of this edge community cloud is one which in part is sustained by user contributions, both in terms of infrastructure and in terms of maintenance. Similar to the file sharing applications in P2P networks, users need not to be only consumers of services, but have also the possibility to become service providers.

### III. RELATED WORK

Several elements of edge computing and collaboration, which we touch upon in presenting our approach, have been developed successfully before.

The willingness of people to collaborate on computing tasks has been demonstrated by volunteer computing, such as in Folding@home [5], which gained a large number of participants to donate computing resources to support biomedical research. Collaborative edge clouds share the same concept that users would be willing to donate computing resources from their personal devices to the community. We notice, however, that the tasks to be performed in volunteer computing are defined and accepted beforehand, which may strongly affect the participants' support. On the contrary in edge community clouds, tasks are less defined and the value of donating the resources may need to be explained to attract users.

Provisioning container-based services using customer premises equipment (CPE) was studied by network equipment vendors [6] and practical implementations showed the technical readiness of the approach [7]. The usage of Docker has also been shown for the ARM architectures [8], which is used in many of the popular single-board-computers (SBC) such as the Raspberry Pi. We are also employing Docker containers in our proposed approach to the collaborative edge clouds. One of the differentiating requirements in our scenario is the ease of management of the Docker-based services for an average end user, for which we need to put emphasis on a graphical container management interface.

Data processing for the IoT domain, by conducting data-intensive computing on distributed and decentralized clouds at the network edge [9], is another area under active research. To perform such data operations, microclouds formed by edge devices were presented recently [10]. Our approach similarly considers low-capacity devices to form edge clouds, but our scenario corresponds less to a local homogeneous cluster, and applies more to the geographically distributed, loosely coupled heterogeneous devices.

Personal cloud services, which aim to provide customized cloud service experience to the users, such as increased privacy, are beginning to get market share, as shown by the Nuvlabox [11]. This solution provides an environment to poll remote images for installation on powerful small form factor PCs located at the users' homes. Similar to this work, in our scenario we consider the edge device to be also available for personal services. In addition, though, we extend the usage of the edge devices to contribute to services (such as in

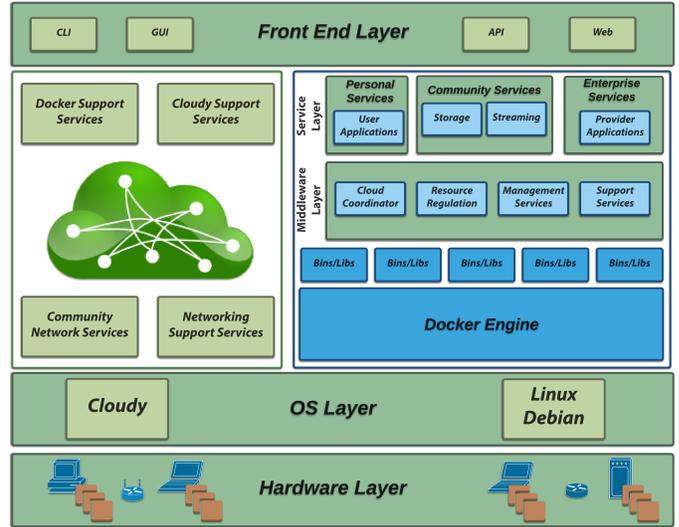


Fig. 1. Layered architecture for Cloudy platform with Docker

volunteer computing), and seek also to enable their usage for commercial services, for instance IoT data storage, integration and analytics.

### IV. ARCHITECTURE

We describe in the following a layered architecture (Figure 1) for a collaborative edge computing platform, which from the bottom to the top layer consists of the following components:

1) *Hardware Layer*: The hardware layer provides the physical infrastructure including the computing and storage devices, and the network equipment required for running the cloud services and applications.

2) *OS Layer*: The operating system layer consists of Cloudy distribution [12], which is based on Debian GNU/Linux. The OS layer is responsible for managing the hardware as virtualized resources. There are a number of support services that are run on top of Cloudy:

- 1) *Community networking services* consists of applications necessary for the operation of the community network.
- 2) *Networking support services* consists of networking-related applications.
- 3) *Docker support services* help with management, deployment and operation of Docker containers, including components such as controller, scheduler, monitor, and data storage for the containers.
- 4) *Cloudy support services* consists of components, such as a manager for the hosts and the network, and can include other components such as: (a) network coordination component to identify and manage different local clouds; (b) service discovery component to keep track of the services provided by the various clouds; (c) authentication and auditing components to support resource regulation; among many others.

3) *Docker Layer*: The Docker engine in Cloudy runs on the host operating system, and includes the necessary

binary packages and libraries for executing applications within containers.

4) *Middleware Layer*: The middleware layer brings together the resources from multiple edge devices, providing an integrated and consistent view of the cloud system to the cloud services. This can comprise of a variety of support services:

- 1) *Cloud coordinator* and services broker components for assisting in combining resources from multiple cloud providers.
- 2) Incentives-based *resource regulation* and allocation components [13] are important as a driver for users' participation for the sustainability and growth of the community cloud model. This can also include a *trusted auctioneer* module for auction-based resource allocation schemes [14]. The resource allocation component needs to be trusted by the users, and the users need to follow the prescribed policies for the system to function properly.
- 3) *Management services* provide tools to assist cloud applications, and can offer features like authentication, auditing, search, directory lookup, etc.
- 4) Other *support services* can be useful for the operation of the community cloud system, and assist in designing, deploying and consuming cloud applications.

5) *Services Layer*: The Services layer integrates useful services and applications [15] for the community cloud members to encourage their participation. These services are logically grouped into personal services, community services and enterprise services. *Personal services* refer to applications in the Cloudy device which a user runs for private purposes, and which are not shared with the community, e.g. a local storage application. *Community services* represent services which the user runs on the local device to support the community cloud operation, such as a search service. *Enterprise services* are meant for commercial providers for provisioning proprietary services. The owners of the devices can agree to install such proprietary commercial services in order to either personally benefit from a commercial service, e.g. from the fields of health technology, security and surveillance, energy management, or to support a third-party proprietary commercial service, potentially agreed on a contractual basis.

6) *Front-end Layer*: The front-end layer provides the different types of interfaces to interact with the infrastructure of the community cloud, and any other tools for assisting in the development of cloud services and applications.

## V. PROTOTYPE COMPONENTS

We explore the implementation of our proposed architecture through the Cloudy<sup>1</sup> distribution [12], which has been developed to build community clouds. Cloudy is installed by the users on their edge devices. Through the support services integrated in Cloudy, the edge node can then become part of the community cloud. Figure 2 shows a typical device<sup>2</sup>, that is part of such a collaborative edge cloud.

<sup>1</sup>Cloudy is available at <http://cloudy.community>

<sup>2</sup>NEO Z64 Mini PC: <http://minix.com.hk/en/products/neo-z64-windows>



Fig. 2. Device used to build the edge community clouds

### A. Services from the IoT Domain

We are interested to evaluate the use case of a large commercial services provider that offers software solution for IoT, distributing it both as a commercial enterprise edition and an open-source community edition, in order for us to test some prototype components.

For this purpose, we have chosen Hitachi Vantara's Pentaho open-source business intelligence (BI) suite [16], that provides comprehensive reporting, data processing, data integration, and data mining features. The Pentaho Data Integration (PDI) client is a desktop application that users can use to process and integrate data on their own machines, and design data workflows that they can upload to the repository in the Pentaho Server. Users can schedule jobs to run data workflows at regular intervals, and perform other advanced operations. The Pentaho Platform also provides web-based tools to create reports, dashboard and charts in a browser. Pentaho also includes Weka, a popular open-source data mining software, and also integrates support for executing machine learning programs through its integration of Python and R. Pentaho provides an extensible plugin architecture, so users can integrate support for their special use cases. For these services from Pentaho platform, there is also the option to install them as Docker containers, as we explain in the following sections.

### B. Deployment Environment

The environment we use for exploring the collaborative community cloud is a real deployment in Guifi.net [15], which is operational since 2015. Guifi.net is a community network located in Barcelona and other areas of Spain. It has more than 30,000 operational nodes and can be considered the largest community network worldwide [17]. The community cloud in this case consists of distributed heterogeneous computing devices contributed by the network participants. Such devices are typically inexpensive mini-PCs or virtual machines (VMs), which run the Cloudy software [12].

### C. Practical Experimentation

In our setup using Cloudy in Guifi.net, our goal is to deploy the Pentaho Server as a Docker container on the nodes in the edge community cloud, which are running Cloudy software. In our setup, we consider the scenario where each owner of the

%	Description	Host	IP	Port	µcloud	Action
	wordpress:latest_testwordpress_wordpress_1	clouddev.guifi.local	10.139.40.56	8000		Enter App
	eclipse-mosquito:latest_etcio_stallman	clouddev.guifi.local	10.139.40.56	1893		Enter App
	owncloud:8.1_gracious_cray	clouddev.guifi.local	10.139.40.56	9910		Enter App
	wmarinho/pentaho-biserver:5.1-TRUNK_pentaho	cloudy0eb9CL1a.guifi.local	10.139.40.76	8080	guifi	Enter App
	wmarinho/postgresql:9.3_postgresql	cloudy0eb9CL1a.guifi.local	10.139.40.76	5432	guifi	Enter App

Fig. 3. Pentaho service installations in Cloudy

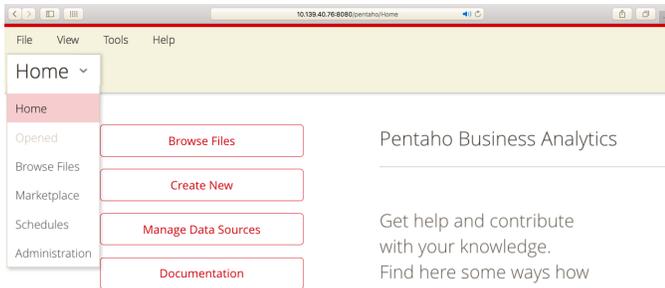


Fig. 4. Access to the Pentaho service installed by the user

node sets up an independent Pentaho Server instance, and the other users of the edge community cloud will contact the owner to issue them login credentials for that instance of Pentaho Server. For data collection and storage, the user can have a virtual machine with a public cloud provider on the Internet, or set up another node in the edge community cloud as an FTP or a database server, where the nodes will upload the data received from IoT devices. In our setup, we have used PostgreSQL database, which also runs as a Docker container on a Cloudy device.

After the installation of Pentaho services in Cloudy, we can observe in Figure 3 from the perspective of a user connected to the Cloudy device with IP address 10.139.40.88 that she can find a Docker-based Pentaho service available in the community cloud. The Pentaho service is available from a provider with IP address 10.139.40.76, while there are other services offered by different providers too.

The Pentaho service is discoverable in Cloudy because the provider has tagged it as a *shared* service. Therefore anyone in Guifi.net running Cloudy software will be able to find it. Figure 4 shows the home screen of a Pentaho service, where the user can manage stored files and scheduled jobs, after logging in by accessing it through Cloudy web interface.

## VI. CONCLUSION

Collaborative edge cloud computing proposes a particular form of edge computing, in which the services and resources of low-capacity home devices are shared among the users. The Cloudy software provides a flexible deployment framework leveraging Docker for provisioning applications. The owners

of the edge cloud nodes can decide on sharing the resources of their devices, which can enable lowering the entry barriers for third-party service providers, and so has the potential to foster the deployment of tailored services at the network edge. The preliminary experiments with services installations demonstrated the concept of this approach. The experience from this deployment suggests future practical work on combining diverse services into an integrated IoT service. The assessment of the characteristics of such a service should provide further insights into the capabilities of the proposed edge clouds.

## ACKNOWLEDGMENT

This work was supported by the European Union's Horizon 2020 research and innovation programme under the project net-Commons (H2020-688768), and the project FIESTA (H2020-643943), and by the Spanish government under contract TIN2016-77836-C2-2-R.

## REFERENCES

- [1] Dan Kirsch, "The Value of Bringing Analytics to the Edge," 2015. [Online]. Available: [http://i.dell.com/sites/doccontent/shared-content/data-sheets/en/Documents/Value\\_of\\_Analytics\\_at\\_the\\_Edge\\_Final.pdf](http://i.dell.com/sites/doccontent/shared-content/data-sheets/en/Documents/Value_of_Analytics_at_the_Edge_Final.pdf)
- [2] "Intel IoT Platform Reference Architecture." [Online]. Available: <http://www.intel.com/content/www/us/en/internet-of-things/white-papers/iot-platform-reference-architecture-paper.html>
- [3] "Eclipse IoT Cloud Platform Stack." [Online]. Available: <https://iot.eclipse.org/>
- [4] W. Shi and S. Dustdar, "The Promise of Edge Computing," *Computer*, vol. 49, no. 5, pp. 78–81, May 2016.
- [5] A. L. Beberg *et al.*, "Folding@home: Lessons From Eight Years of Volunteer Distributed Computing," in *8th IEEE International Workshop on High Performance Computational Biology (HiCOMB '09)*, within *IPDPS*, Rome, Italy, May 2009.
- [6] "Internet of Things (IoT): Transform Data into Action at the Network Edge." [Online]. Available: <http://www.cisco.com/web/solutions/trends/iot/fog-computing.html>
- [7] D. F. Willis, A. Dasgupta, and S. Banerjee, "ParaDrop: A Multi-tenant Platform for Dynamically Installed Third Party Services on Home Gateways," in *2014 ACM SIGCOMM Workshop on Distributed Cloud Computing (DCC '14)*, Chicago, IL, USA, 2014, pp. 43–44.
- [8] A. Eiermann, M. Renner, M. Großmann, and U. R. Krieger, "On a Fog Computing Platform Built on ARM Architectures by Docker Container Technology," in *17th International Conference on Innovations for Community Services (IACS 2017)*, Darmstadt, Germany, Jun. 2017.
- [9] A. Chandra, J. Weissman, and B. Heintz, "Decentralized Edge Clouds," *IEEE Internet Computing*, vol. 17, no. 5, pp. 70–73, Sep. 2013.
- [10] Y. Elkhatib *et al.*, "On Using Micro-Clouds to Deliver the Fog," *IEEE Internet Computing*, vol. 21, no. 2, pp. 8–15, Mar. 2017.
- [11] "Nuvlabox Smart Cloud-in-a-Box Appliance." [Online]. Available: <http://sixsq.com/products/nuvlabox/>
- [12] R. Baig *et al.*, "Deploying Clouds in the Guifi Community Network," in *IFIP/IEEE International Symposium on Network Management (IM 2015)*, Oct. 2015, pp. 113–120.
- [13] A. M. Khan, U. C. Buyuksahin, and F. Freitag, "Incentive-based resource assignment and regulation for collaborative cloud services in community networks," *Journal of Computer and System Sciences*, vol. 81, no. 8, pp. 1479–1495, Dec. 2015.
- [14] A. M. Khan, X. Vilaça, L. Rodrigues, and F. Freitag, "A Distributed Auctioneer for Resource Allocation in Decentralized Systems," in *36th IEEE International Conference on Distributed Computing Systems (ICDCS 2016)*, Nara, Japan, Jun. 2016.
- [15] M. Selimi *et al.*, "Cloud Services in the Guifi.net Community Network," *Computer Networks*, vol. 93, no. P2, pp. 373–388, Dec. 2015.
- [16] M. R. Casters, R. Bouman, and J. van Dongen, *Pentaho Kettle Solutions: Building Open Source ETL Solutions with Pentaho Data Integration*. Wiley, 2010.
- [17] B. Braem *et al.*, "A Case for Research with and on Community Networks," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 3, pp. 68–73, Jul. 2013.