Nomadic Datacenters at the Network Edge: Data Management Challenges for the Cloud with Mobile Infrastructure

[Vision Paper]

Faisal Nawab
UC Santa Cruz
Santa Cruz, CA
fnawab@ucsc.edu

Divyakant Agrawal, Amr El Abbadi
UC Santa Barbara
Santa Barbara, CA
[agrawal,amr]@cs.ucsb.edu

ABSTRACT

The continuing growth and success of many edge technologies such as the Internet of Things (IoT), wearables and Virtual and Augmented Reality (VR/AR) relies on providing a high-performance and low-latency computing infrastructure. In this paper, we envision extending edge computing with mobile, moving, and possibly flying edge datacenters, that we call *nomadic datacenters* to improve the performance and capacity of the edge infrastructure. In particular, we study how the introduction of nomadic datacenters will affect data management systems and find that novel challenges and opportunities need to be addressed. We present some of these challenges and opportunities in addition to an outline of how they can be tackled by future data management systems.

1 INTRODUCTION

Emerging classes of computing technologies are promising to transform our lives, change how we interact with each other and with the world. These include Internet of Things (IoT), wearables, and Virtual and Augmented Reality (VR/AR). IoT enables harnessing the multitude of sensor data via applications ranging from smart farming to autonomous cars. Wearable technology enables personalized applications such as activity tracking and health monitoring. With VR/AR, we will be immersed in designed experiences that will touch every facet of our lives. Common among these transformative technologies are the utilization of sophisticated edge devices and the demand for high-throughput and/or low-latency. We will use the term *edge technologies* to denote IoT, wearable, and VR/AR technologies in light of their common characteristics.

To realize the potential of edge technologies, it is necessary to provide the hardware and software infrastructures that support the high-throughput and low-latency demands for application processing. To this end, many efforts have advocated for the use of edge computing technology to provide more compute and storage power [4, 13]. Edge computing enhances cloud computing by introducing *edge datacenters* that are closer to users and that consist of a few to hundreds of machines. With edge computing, data can be processed at the edge, saving the communication latency to the datacenter that can take up to 100s of milliseconds [13]. Additionally, processing at edge datacenters saves the monetary bandwidth costs of cloud-edge communication.

Realizing the benefits of edge datacenters has been limited by the static nature of edge datacenter deployment. Typically, edge datacenters are rigidly stuck in fixed locations. This introduces two limitations:

- *A static deployment cannot follow the hot spots:* It is difficult for a static deployment to adapt to a dynamic, mobile environment, where the current location of the highest data traffic is continuously, and often unpredictably changing. For example, consider building an edge infrastructure to support a taxi transportation organization. Taxi cabs connect to edge datacenters that are placed in various locations around the city. However, the location of taxi cabs depends on many volatile aspects such as traffic and passengers. This makes it very difficult to decide where to place the edge datacenters and how to provision resources around the city.
- *A static deployment cannot be recovered swiftly:* In the event of a natural disaster and power outages, a large area may be affected. The edge datacenters across these large areas may be inaccessible or even permanently damaged. To recover from such catastrophic failures that damage infrastructure, replacing the infrastructure is necessary. However, it may take days to replace and deploy the new infrastructure. This is especially devastating when the edge infrastructure is needed to aid in responding to natural disasters.

In this paper, we propose extending edge computing technology with dynamic, mobile edge datacenters, and call them **nomadic datacenters**. Nomadic datacenters denote small, portable edge datacenters, that can be relocated swiftly by large vehicles (*e.g.*, trucks) and air crafts (*e.g.*, helicopters). Nomadic datacenters can overcome the two limitations of static edge datacenters that we outlined in the previous paragraph. A nomadic datacenter *can* follow the hot spot. In the taxi organisation example, nomadic datacenters can be continuously moving to maximize the utility of resources. Also, nomadic datacenters can replace a damaged infrastructure swiftly in cases of natural disasters and power outages. In fact, nomadic datacenters can be thought of as an aid to first responders that may need the edge computing resources to collect and process data in addition to providing connectivity during relief and rescue operations.

The potential feasibility of the concept of nomadic datacenters is due to recent advances in datacenter and communication technology. Edge datacenter technology has been continuously improving during the past decade, resulting in small, containerized datacenters, also called *micro datacenters.* A micro datacenter may contain as little as a few servers on a single rack with built-in cooling, power supply/backup, and fire suppression systems.

Published in Proceedings of the 21st International Conference on Extending Database Technology (EDBT), March 26-29, 2018, ISBN 978-3-89318-078-3 on OpenProceedings.org.

Series ISSN: 2367-2005

10.5441/002/edbt.2018.56

Already, these micro datacenters are being leveraged for their portable nature that allows deploying them in remote areas such as shallow- and deep-water oil rigs [1]. Our proposal is to leverage this portability to react to dynamic, mobile edge applications and actively follow hotspots in addition to replacing damaged edge infrastructure.

The other technology that is enabling nomadic datacenters is 5^{th} generation mobile networks (5G). 5G is 4G's successor telecommunication standard. It aims to provide lower latency, higher communication throughput, and low power transmission. More importantly, 5G is geared towards supporting emerging edge applications, such as ones based on IoT and wearables. This is envisioned by providing support of device-to-device and mobile broadband communication. This is significant for the realization of nomadic datacenters that will rely on wireless communication to connect to edge users/devices and the cloud. 5G will allow larger-scale communication between a nomadic datacenter and edge users and devices through device-to-device and mobile broadband communication. Also, nomadic datacenters would need a large capacity wireless link to connect to the backbone in the cloud. 5G's larger capacity will ameliorate the capacity limits of wireless telecommunication technology compared to wired and optical fiber communication. Also, nomadic datacenters would typically be powered for long (or all) durations on batteries. Thus, low power consumption based on 5G is an important feature.

The concept of a nomadic datacenter is not new, as many have suggested the idea for its practical uses in many applications [12]. However, it was not realized because the relevant communication and datacenter technologies were not ready. With the advances of micro datacenters and 5G telecommunication, nomadic datacenters are positioned to be a reality now more than ever.

In this paper, we envision the last piece in the puzzle of making nomadic datacenters a feasible technology—*we propose the study of data management systems for nomadic datacenters*. In addition to the hardware and communication infrastructures (*i.e.*, micro datacenters and 5G) to realize nomadic datacenters, data management technology must be revisited to tackle the unique challenges and exploit the opportunities of the new edge architecture. We present a system model that encompasses several scenarios of how nomadic datacenters will be realized. Then, our study of the system model reveals a set of novel data management challenges and opportunities. We present these challenges, opportunities and a roadmap to tackle them.

2 A SYSTEM MODEL FOR NOMADIC DATACENTERS

In this section, we present our vision of emerging system models of nomadic datacenters. The development of these system models is important to guide the design decisions and identification of the salient properties of the new technology. We begin by describing the base architecture of nomadic datacenters and its interaction with the cloud, users, and other edge datacenters. Then, we discuss how the model can be adapted to various properties of nomadic datacenter deployments that have varying sizes and mobility characteristics.

The base system model consists of three tiers (shown in Figure 1):

- *(1) Cloud tier:* This denotes the cloud resources at traditional, large datacenters.

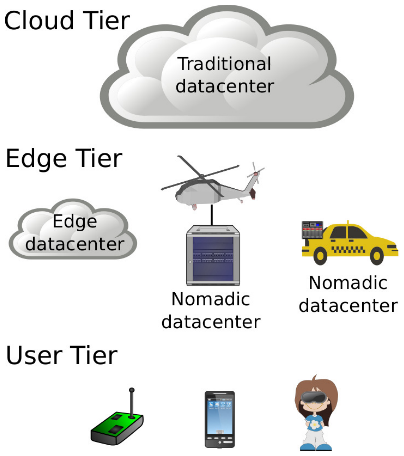

Figure 1: The system model with nomadic datacenters.

- *(2) Edge tier:* This denotes the resources at edge and nomadic datacenters. Edge datacenters communicate with the cloud tier through high-bandwidth links, such as fiber optics, and nomadic datacenters communicate with the cloud tier via wireless telecommunication (*e.g.*, 5G). However, nomadic datacenters are also capable of communicating with other edge datacenters through wireless links. This will enable coordination between nodes in the edge tier and also enables relaying communication between nomadic datacenters and the cloud tier through intermediate edge datacenters.
- *(3) User tier:* This denotes the users and devices generating data and making requests. Nodes in the user tier, that we will call user nodes, communicate with the application through the edge tier, if an edge or nomadic datacenter is nearby. Typically, this communication would be through wireless links, using Wi-Fi technology. If no edge or nomadic datacenter is nearby, then the user nodes communicate directly with the cloud tier.

Nomadic datacenters will vary in size to adapt to various application and environment requirements. We abstract the different sizes of nomadic datacenters to fall in one of three sizes: (1) Light: This represents nomadic datacenters that contain a single machine and minimal datacenter capabilities for cooling and security. This is ideal in cases where the nomadic datacenter should be carried by small vehicles, such as drones. (2) Medium: This represents nomadic datacenters that contain a few machines and is ideal to be carried by small vehicles such as taxi cabs. (3) Heavy: This represents nomadic datacenters that resemble current micro datacenters that contain 4 or more machines with various datacenter capabilities. This is ideal for cases where a truck or an aircraft carries the datacenter. The size of the nomadic datacenter influences its mobility characteristics. For example, light nomadic datacenters can be deployed for high mobility scenarios with constant movement and relatively smaller vehicles (*e.g.*, drones). Medium ones can be deployed on medium-sized vehicles, and thus can be used for mobility cases in urban settings. Heavy nomadic datacenters has relatively restricted mobility and are ideal in cases where mobility is in reaction to a non-frequent event.

Given this view of the system model of nomadic datacenters, we discuss some data management issues in the context of nomadic datacenters.

	Centralized Infrastructure	**Extended Infrastructure**
Static Resources	Client-Server (Cloud Computing)	Edge Computing
Mobile Resources	MDMS	Nomadic Datacenters

Table 1: The high-level differences in the system model of Nomadic datacenter in comparison to other system models.

3 DATA MANAGEMENT FOR NOMADIC DATA CENTERS

In the nomadic datacenter architecture, is there a need for innovation in data management systems to support the new environment or do existing systems suffice? This section answers this question by discussing how the nomadic datacenter architecture is positioned in relation to early work on mobile databases (Section 3.1). Then, we show how the unique properties of nomadic datacenters require innovation in data management systems in Section 3.2.

3.1 Nomadic Datacenters in the Space of Mobile Data Management

Building data management systems for mobile environments has been studied extensively for a number of decades [3, 6, 8, 9]. In these early works, that we will denote as *mobile data management systems* (MDMS), users and data copies are mobile and may use wireless communication. These properties are similar to the properties brought forth in the nomadic datacenters system model (Section 2).

So, can we just use MDMS [3, 6, 8, 9] to build solutions for nomadic datacenters? Our study of MDMS systems have revealed that they are an excellent starting point and basis for data management solutions for nomadic datacenters. MDMS outlines many solutions to tackle challenges that also arise in nomadic datacenters such as resources asymmetry, mobility, caching, and energy efficiency. Many of these techniques can—and should—be adopted by data management systems for nomadic datacenters.

However, there is a key difference in the nomadic datacenters architecture compared to early MDMS architectures. In MDMS, there is a "core" database and mobile users. This makes MDMS consider a centralized infrastructure with mobile resources. For nomadic datacenters, the users are mobile like MDMS, but the infrastructure is different. The infrastructure in nomadic datacenters is extended beyond a centralized location to mobile edge nodes (*i.e.*, nomadic and edge datacenters).

Table 1 summarizes the high-level position of nomadic datacenters compared to relevant architectures. An interesting relation that can be observed from the table is that *a nomadic datacenters architecture to MDMS is what edge computing is to cloud computing*. Therefore, the data management challenges of edge computing in comparison to cloud computing will likely exist for nomadic datacenters in comparison to MDMS. The nature of these challenges stems from the fact that edge resources are more powerful than client machines, and thus there is an opportunity to leverage them more aggressively for data management tasks like caching [5], and offloading [11, 14]. This makes the caching and offloading results for edge computing [5, 11, 14] different from ones for MDMS [3, 6, 8, 9].

Although transforming edge solutions to adapt to the nomadic datacenter architecture is important and may entail interesting designs, we are interested more in the novel challenges and opportunities that are unique to nomadic datacenter architecture. The next section introduces such challenges and opportunities.

3.2 New Challenges and Opportunities

We postulate that the nomadic datacenter architecture has fundamental differences compared to previous architectures and combinations of previous architectures (*e.g.*, MDMS with edge computing). These differences require innovative solutions to tackle the challenges and benefit from the opportunities of nomadic datacenters. In this section, we present the imminent challenges and opportunities in this space.

3.2.1 The wireless link: a bottleneck and an opportunity. Nomadic datacenters communication with users, other edge and nomadic datacenters, and the cloud tier through wireless links. Although emerging technology such as 5G will ameliorate this bandwidth limitation of wireless links, they are still limited compared to wired infrastructures. In addition, communication between nomadic datacenters and users and other edge datacenters will likely still rely on Wi-Fi or similar technologies. Therefore, *communication bandwidth will be an extremely costly resource for nomadic datacenters.* This is not the case for other architectures. (Unlike MDMS, a nomadic datacenter manages the data of a large number of users and thus require a significantly larger bandwidth than a single MDMS client.) The bandwidth cost significantly affects the design trade-offs space for data management tasks as we outline in the next example.

A case study on coordination. An example of a data management task that will be affected by the high bandwidth cost is coordination. Coordination is necessary to maintain consistency across different copies of data via protocols such as Two-Phase Commit [7], Paxos [10], and others. Coordination is communication-intensive, where each request is typically coordinated among nodes that hold copies of the accessed data. Take for example a scenario of two nomadic datacenters, *A* and *B*, that are located to be close to a large event that is anticipated to generate a lot of traffic. Both nomadic datacenters, *A* and *B*, are providing access to the same data but half the users are connected to *A* and the other half is connected to *B*. Now, in applications where requests would require coordination (such as OLTP transactions), *A* and *B* must coordinate every request they receive. For example, a request that is received at *A* would create a coordination request from *A* to *B*. This means that, potentially, double the wireless link bandwidth is consumed than necessary. This is exacerbated for cases with more nomadic datacenters and represents a major source of wasted resources and stress to the most limited nomadic datacenter resource—communication bandwidth.

Optimizing the bandwidth of coordination is an open challenge for nomadic datacenters and can be tackled based on familiar approaches in batching and compression. However, there are also opportunities for innovative solutions to this problem that are allowed by the unique architecture of nomadic datacenters. Specifically, since all communication is through the wireless medium, *nodes can eavesdrop on other nodes' communication.* Consider our earlier coordination scenario where *A* receives a request and then coordinate with *B*. Since *B* can eavesdrop on the communication from the user to *A*, then it already knows about the request, even before *A* initiates coordination for it. This allows efficient alternatives to coordination. For example, *B* can signal that it has heard about the request and decides a certain action regarding it. This is more efficient in two ways: (1) *A* does not have to communicate with *B*, and (2) the communication from *B* is a control message only and does not need to contain the payload of the request that is likely much bigger than control messages. Another example to reduce coordination overhead is partitioning. However, unlike traditional partitioning where the system controls partitioning the data only, in our scenario, the nomadic datacenters can also partition the users between them. This means that *A* and *B* can judiciously decide which users connect to which nomadic datacenter, and in jointly with a data partitioning strategy between *A* and *B*, the need for coordination between *A* and *B* would be reduced.

To summarize, we foresee innovations in coordination and other data management tasks that tackle the main bottleneck of nomadic datacenters (communication bandwidth) via innovative solutions specific to the nomadic datacenter environment such as eavesdropping and users-data partitioning.

3.2.2 Challenges of a Dynamic Mesh Environment. In many of the scenarios we envision, there is a large number of nomadic datacenters that roam around continuously. For example, consider a deployment on taxi cabs, where each taxi cab contains a nomadic datacenter. This may serve an application for the taxi cabs, but may also be serving urban applications. In this scenario, there are potentially thousands of nomadic datacenters that get connected to each other sporadically. We call this a *dynamic mesh model*, akin to Wireless Mesh Networks (WMNs) [2]. However, unlike WMNs, mobility and relocation are rapid and each node is a nomadic datacenter that performs extensive computation and performs data management tasks on behalf of users, rather than just being a communication hub.

The dynamic mesh model introduces serious challenges to distributed protocols that are necessary to manage replicated or partitioned data across nodes. Such distributed protocols, such as Paxos [10], Two-Phase Commit [7], and others, were designed for a static infrastructure, where the participants in a protocol are known. However, in a dynamic mesh model, the participants and topology configuration changes continuously, *requiring the invocation of expensive membership and leader election protocols continuously.* This motivates the study of data management system designs that assume that membership and leader election is invoked frequently.

A case study on Paxos. For example, consider the Paxos protocol [10]. Paxos performs two tasks: leader election and replication. Replication is performed by a leader to make sure data is persistent. Leader election is only invoked during a suspected failure of the current leader. Therefore, many Paxos designs and variants optimize for the case where leader election is rare. Furthermore, Paxos reconfiguration is invoked in cases where a machine is to be replaced or to migrate the infrastructure. Paxos reconfiguration [10] is very expensive and in traditional deployment this is not problematic because it is extremely rare. Because nomadic datacenters move rapidly and the configuration changes continuously, leader election and reconfiguration are invoked frequently. This invites a redesign of the Paxos protocol to make leader election and reconfiguration more efficient. Such redesigns might be enhanced by exploiting the opportunities enabled by the special characteristics of nomadic datacenters. An example is that leader election and reconfiguration are triggered by mobility in nomadic datacenters, rather than failures as the traditional cases. This can be exploited because the current leader is aware of the anticipated leader election and/or reconfiguration and can participate in it. Therefore, rather than elaborate complex mechanisms to ensure the correct election of a new leader, a live, mobile leader can simply relinquish leadership to another node unilaterally.

Although we discussed a single case study on Paxos, the challenges of membership and reconfiguration in an extremely mobile environment are applicable to a wide-range of distributed protocols. We believe that opportunities to optimize membership and reconfiguration mechanisms for nomadic datacenters exist akin to the high-level optimization that we discussed for the Paxos case study.

4 CONCLUSION

Nomadic datacenters have the potential of enabling a wide-range of edge applications that rely on emerging edge technologies such as IoT, wearables, and Virtual and Augmented Reality. Nomadic datacenters introduces a dynamic, mobile infrastructure that allocate resources on demand in reaction to changes in workload and failures. This makes it suitable for emerging mobile edge applications. In this paper, we outline our vision of nomadic datacenters, how they are becoming more feasible, how they will be realized, and what are the imminent data management problems that arise with their introduction. We find that the areas that require attention in designing data management systems for nomadic datacenters are: the mobility of the extended infrastructure hierarchy, the wireless nature of communication, and the dynamic, mesh nature of the topology. We outline pathways to solutions to these problems and envision that they will provide a building block towards realizing nomadic datacenters.

ACKNOWLEDGEMENTS

This work is supported by NSF grant CSR 1703560.

REFERENCES

[1] 2017. How Microsoft is Extending Its Cloud to Chevron's Oil Fields. http://www.datacenterknowledge.com/microsoft/how-microsoft-extending-its-cloud-chevron-s-oil-fields. (2017).

[2] Ian F Akyildiz, Xudong Wang, and Weilin Wang. 2005. Wireless mesh networks: a survey. *Computer networks* 47, 4 (2005), 445–487.

[3] Daniel Barbara and Hector Garcia-Molina. 1993. *Replicated data management in mobile environments: Anything new under the Sun?* Technical Report. Stanford.

[4] Flavio Bonomi, Rodolfo Milito, Jiang Zhu, and Sateesh Addepalli. 2012. Fog computing and its role in the internet of things. In *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*. ACM, 13–16.

[5] Ittay Eyal, Ken Birman, and Robbert van Renesse. 2015. Cache serializability: Reducing inconsistency in edge transactions. In *Distributed Computing Systems (ICDCS), 2015 IEEE 35th International Conference on*. IEEE, 686–695.

[6] Jim Gray, Pat Helland, Patrick O'Neil, and Dennis Shasha. 1996. The dangers of replication and a solution. *ACM SIGMOD Record* 25, 2 (1996), 173–182.

[7] Jim Gray and Andreas Reuter. 1992. *Transaction processing: concepts and techniques*. Elsevier.

[8] Joanne Holliday, Divyakant Agrawal, and Amr El Abbadi. 2002. Disconnection modes for mobile databases. *Wireless Networks* 8, 4 (2002), 391–402.

[9] Tomasz Imielinski and BR Badrinath. 1994. Mobile wireless computing: Challenges in data management. *Commun. ACM* 37, 10 (1994), 18–28.

[10] Leslie Lamport. 1998. The Part-time Parliament. *ACM Trans. Comput. Syst.* 16, 2 (May 1998), 133–169. https://doi.org/10.1145/279227.279229

[11] Yi Lin, Bettina Kemme, Marta Patino-Martinez, and Ricardo Jimenez-Peris. 2007. Enhancing Edge Computing with Database Replication. In *Proceedings of the 26th IEEE International Symposium on Reliable Distributed Systems (SRDS '07)*. IEEE Computer Society, Washington, DC, USA, 45–54. http://dl.acm.org/citation.cfm?id=1308172.1308219

[12] B. Rimler and N. Rasmussen. 2006. Mobile data center. (April 20 2006). https://www.google.com/patents/US20060082263 US Patent App. 11/251,587.

[13] Mahadev Satyanarayanan, Paramvir Bahl, Ramón Caceres, and Nigel Davies. 2009. The case for vm-based cloudlets in mobile computing. *IEEE pervasive Computing* 8, 4 (2009), 14–23.

[14] Hemant Saxena and Kenneth Salem. 2015. EdgeX: Edge Replication for Web Applications. In *Cloud Computing (CLOUD), 2015 IEEE 8th International Conference on*. IEEE, 1041–1044.