ISSN 2347 - 3983
Volume 8. No. 6, June 2020
International Journal of Emerging Trends in Engineering Research
Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter32862020.pdf
https://doi.org/10.30534/ijeter/2020/32862020

A Resource-aware Service Discovery Architecture for Ad-hoc Mobile Cloud

Dominic Afuro Egbe[1], **Bethel Mutanga Murimo**[2]
[1]Department of Information Technology, Durban University of Technology, RSA, domafuro@gmail.com
[2]Department of Information and Communication Technology, Mangosuthu University of Technology, RSA, mutangamb@mut.ac.za

ABSTRACT

The synergy between Mobile Cloud Computing and the proliferation of smart devices is increasingly expanding the interaction space for mobile web services delivery. More so, with the rising prominence of the IoT paradigm, which seeks to drive a smarter and unlimited interconnected interaction between diverse devices, the need for opportunistic and inexpensive service provisioning platforms becomes more imperative. Such platforms stand to potentially contribute toward realizing the Ad-hoc Mobile Cloud model which advocates the idea of mobile web services (MWS): enabling mobile and other smart devices to function as service providers in addition to their conventional role as service requesters. However, mobile, and smart devices are inherently characterized by limitations in resources such as battery power, memory, and storage capacities. Consequently, executing service discovery operation on the local device can introduce a huge resource burden. Therefore, to achieve service discovery in such a decentralized mobile device-enabled e-services provisioning platform without compromising the devices' resources requires an architecture designed to support capabilities for proactive resource management. In this paper, we propose an architecture that supports service discovery in Ad-hoc Mobile Cloud (AMC) through a mechanism driven by resource-awareness.

Key words: Architecture, Ad-hoc mobile cloud, Mobile cloud, Personalization, Service discovery, Web services

1. INTRODUCTION

Mobile Cloud Computing (MCC) has, in the last decades massively revolutionized the cloud service provisioning by building upon the enormous benefits of conventional Cloud Computing [1], [2]. Fundamentally, MCC integrates Cloud Computing technological concepts and mobile devices to drive multiple platform support for service delivery [3]. Through this hybrid model, MCC specifically adds to the benefits of Cloud Computing by boosting the capability to guarantee consistent availability of services anywhere and at any time [4], while reducing or eliminating the need for hardware equipment [3]. With this capability, mobile cloud service provisioning has grown in prominence as it requires almost zero infrastructure for cloud service providers to offer diverse services that are typically composed and invoked from mobile devices.

Although MCC guarantees that cloud services are available in the context of making them accessible anywhere and at any time from remote servers, the core aspect of service availability lies in the discoverability of such services. This discoverability condition requires cloud service provisioning platforms to implement service discovery mechanisms that enable consumers to not only discover services but services that are relevant to their current context, that is the most preferred services. Furthermore, given the fact that users often have varying preferences, coupled with the diverse abundance of mobile devices in use, service discovery processes must enable the discovery of relevant services. This requirement is the central idea behind the concept of personalization [5]. Personalization which is about implementing mechanisms that can determine the suitability of a given cloud service to its consumer at the discovery stage has put context-awareness at the centre stage of research in pervasive environments [6], which focuses also on architectural evolution.

While MCC presents a huge and multi-dimensional prospect, the emerging trends in the domain further introduce an interesting and challenging opportunity to broaden the cloud-service provisioning and discovery landscape. These trends from an economic viewpoint highlight the growing adoption of MCC as the dominant driver of global businesses in the 21st century. This is because the MCC's business model is characterized by unique flexibility that has in recent years catalyzed the rapid growth of Small, Medium & Micro Enterprise Businesses (SMMEs) by providing them with the leverage to engineered business diversity [7]. This, to a large extent, explains the trend of the rising trajectory of MCC adoption worldwide in the last decade [8], as well as the promising forecasts. For instance, in the report "Mobile Cloud Market - Growth, Trends, and Forecast" [9], Mordor Intelligence states that "the mobile cloud market was worth USD 30.71 billion in 2019 and that it is expected to reach USD 118.70 billion by the end of 2025".

Among others, two major driving factors are key to the growth rate of MCC adoption and fascinating forecasts. First, the explosive growth in the number of mobile devices. As indicated by Badidi et al in [10], the global mobile device user-base has proliferated, with a corresponding increase in the number of MCC subscribers. In fact, as at the beginning of the last decade, the Cisco Internet Business Solutions Group (IBSG) as cited in [11], already maintained that an excess of 80% of the world's population had access to diverse mobile devices. Interestingly too, mobile devices' use has equally evolved from the traditional mobile phone used to store media to tools for accessing media from other devices or for accessing other applications and value-added services offered remotely [11], [12]. This factor coupled with the tremendous rise in Internet usage over the past few years significantly accounts for the increasing adoption of MCC [13]. The second driving factor is the fascinating development of supporting or enabling technologies. There is a progressive evolution of a wide range of state-of-the-art technologies that are expected to radically transform and galvanize the mobile cloud market in the near future. These include technologies such as 5G [14], HTML5, CSS3, Hypervisor [15], Cloudlets [16], and Web 4.0 [17]. For instance, these technologies, among other benefits, will pave way for specifications for offline support, enable an architecture-neutral execution of web applications on any smartphone, and guarantee reduced latency with a faster response.

The Internet of Things (IoT) has been another interesting emerging trend gaining significant research momentum. Basically, IoT in conjunction with mobile communication technologies promises to facilitate the design and deployment of billions of cloud-based mobile sensors and wearable computing devices [18]. Fundamentally, IoT aims at enabling interaction and cooperation between things and objects, which ultimately speaks to the broad nature of the application of IoTs [19].

The overall effect of these trends will be characterized by a continuous surge in the number of cloud services, service provers, and the demand for such services as well. However, this enhancement does not remove the inherent challenge of limited resources associated with mobile devices, and the cost factor effect on mobile service consumption. For instance, traditionally, resource-intensive computations are offloaded to the cloud to relieve mobile devices of the associated energy-burden [18] but the mobile devices still bear the battery draining bandwidth effect. Also, it is infeasible to always have an affordable and reliable Internet connection.

To address this challenge, the Ad-hoc Mobile Cloud (AMC) paradigm serves to complement MCC by "enabling a platform for inexpensive and collaborative" [20] mobile service provisioning where services are co-hosted and discovered among peer devices when the main cloud is inaccessible [21]. However, the unique nature of the AMC and the envisaged role of mobile peers also present unique service discovery challenges that cannot be suitably addressed by conventional MCC solution approaches. For instance, Stergiou [3] stated that "cloud techniques are generally energy-unaware and bandwidth-hungry", which implies that such techniques may not be suitable for AMC due to their huge resource requirement. This paper, therefore, proposes an architecture that supports resource-aware service discovery in AMC.

2. A CASE FOR RESOURCE-AWARENESS

Ubiquitous environments are inherently dynamic because their constituents are primarily mobile and changing. This inherent nature coupled with the diversity of mobile devices poses a huge service discovery challenge – user preferences are not stable, devices' capabilities vary – therefore, the quality of service may not be guaranteed. Research to address this challenge has inspired the concept of context-awareness, which is about how to make the service discovery process attuned to variations in users' settings. The central goal of context-awareness is captured in Hagen's definition of personalization [22]: "the ability to provide content and services tailored to individuals based on the knowledge about their preferences and behavior".

An efficient service discovery mechanism, therefore, must satisfy a user service request on the fly by returning only context-tailored or relevant services. To satisfy this requirement existing solutions have mainly adopted what can be described as a "user-centric" approach: adapting services requests to return relevant services based on the knowledge about the user's profile [23] which includes user preferences [9], current activities, physical location, and surrounding objects [24].

However, Badidi [10] has argued in support of the need for a more comprehensive approach towards service personalization. According to the author, in the context of MCC, service personalization needs to be driven by three components of context informanation, namely, "the user's functional needs", "the device in use", and "the user's context". We extend on Badidi's argument to state that personalization in AMC requires a "device-centric" approach where devices context is seen as a crucial determinant of service relevance. Such an approach as illustrated in Figure 1 serves advocates the use of two components of device context to drive personalization in AMC: i) the use of device resources as key context information to ensure that every operation is resource-sensitive in order not to compromise the constrained resources of peer devices ii) consider device preferences as another aspect of device context. The idea is that a service is judged relevant if it meets the current resource state (battery power, memory) and preferences of the consuming device. Moreover, as IoT integrates with MCC, it is expected that some AMC peers can be autonomous devices that directly invoke web services. As such, unlike the conventional user-controlled service invocation, when services are invoked autonomously, only the device's context and preferences are of significance in determining service relevance.

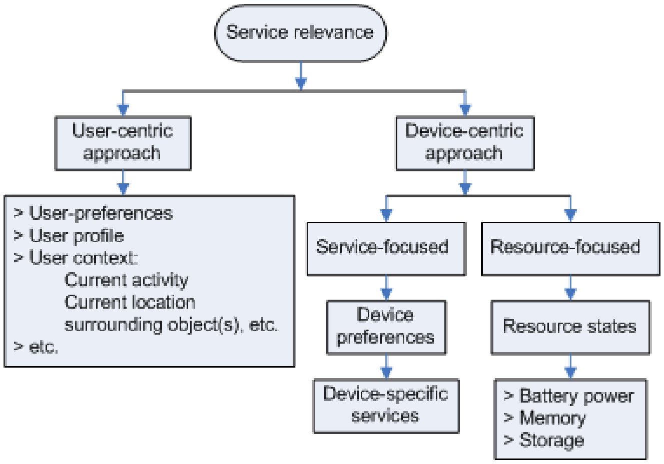

Figure 1: Service Relevance Model For AMC

3. PROPOSED ARCHITECTURE AND COMPONENT DESCRIPTION

The AMC, like every environment, is characterized by resource constraints. This weakness makes the deployment of conventional web service discovery techniques infeasible because they are generally resource-intensive. As such, our proposed architecture is designed to support a resource-aware service discovery that is enhanced by a lightweight service description based on the WSDL-M standard [25]. WSDL-M avoids the use of semantics in service descriptions, which is generally energy-hungry and yet allows the inclusion of both non-functional and functional parameters into web service descriptions to make them more expressive. Our device-centric approach, therefore, leverages on WSDL-M to incorporate service and device focused parameters as depicted in Figure 1 into service descriptions. The aim is to achieve service personalization that is tailored toward the specific resource capability of client devices.

As shown in Figure 2, the architecture has two layers i) the context generation layer ii) the interaction and processing layer, and three main components, namely, the Node Monitor (NoM), Lightweight Database (LDb), and the Discovery Engine (DiEn). There are also two mediating modules – the Service Request Manager (SRM) and Request Listener (ReL).

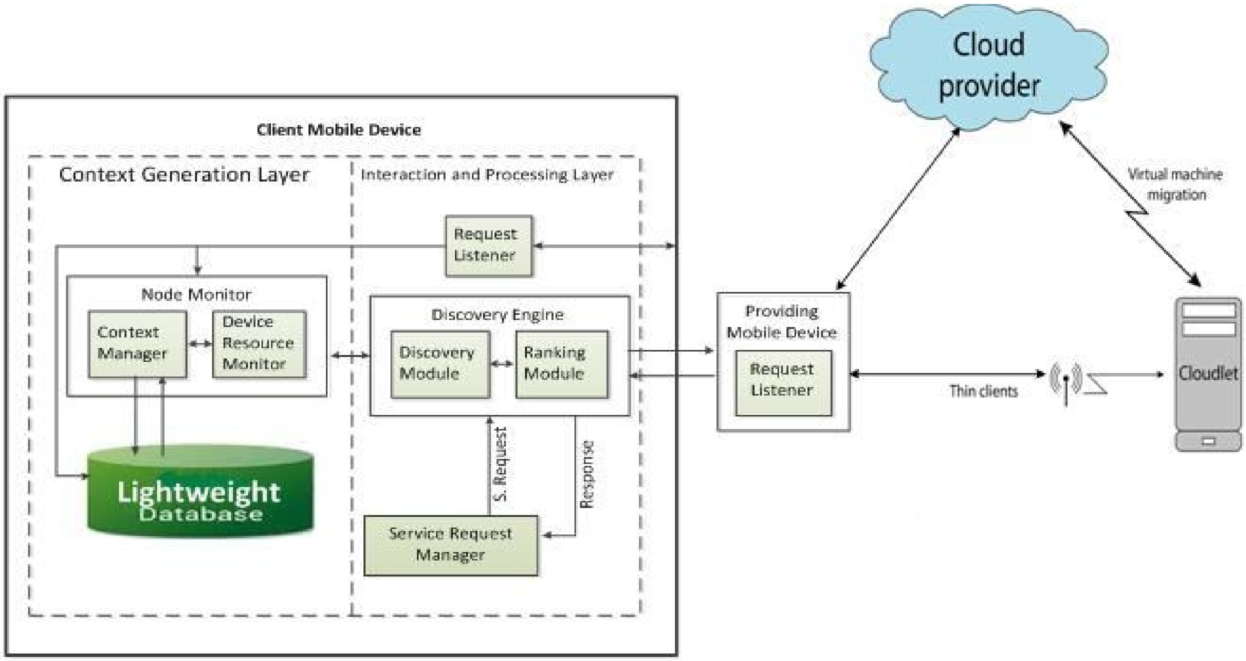

Figure 2: AMC Resource-aware Service Discovery Architecture

In line with our device context idea as discussed in section II, the context and generation layer is made up of components that facilitate the gathering, storing, retrieving, and communicating context information. The Node Monitor (NoM) and Lightweight Database (LDb) are the main components in this layer.

1.1 Node Monitor (NoM)

Our device-centric approached entailed the use of device resources as key context parameters to achieve service personalization or the discovery of relevant services. Such information is dynamically gathered by the NoM component which functions through the interaction between its subcomponents - the Device Resource Monitor (DRM) and Context Manager (CoM).

Device resources are not static. For instance, the available memory may change depending on the current tasks running in the device. Also, battery power changes over time as the running processes use up the stored energy. These changes are tracked by the DRM to provide the current context (current battery level, available memory, and storage) whenever a service request is being processed. For android devices, the Utility functions such as the "FreeMemory()" and "GetBatteryLevel()" can be used to generate this resource context information.

Another vital context information advocated in the device-centric approach is device preferences. Unlike the device resource context, this form of context data is static and is held in the LDb because it is a component of web service specifications. The CoM subcomponent interacts with the LDb and DRM to make both context information available to the NoD.

1.2 Lightweight Database (LDb)

Android devices support an embedded, self-contained lightweight database library with no server component and capability to run a small code footprint, which makes it fit for resource-constrained environments like AMC. That is the rationale for using it as the repository component for the proposed architecture to host web services and static context information.

The second layer of the architecture consists of components that facilitate both interaction and processing. These functions are performed by the DiEn component, supported by two mediating module – the SeL and SRM.

1.3 Service Request Manager (SRM) and Request Listener (ReL)

By design, the architecture support two kinds of interaction: interaction between a service requester and their mobile device (internal) and the interaction between a device that is hosting service and another peer device requesting the service (external). Service requests are constructed through the Service Request Manager (SRM) which provides an interface for service queries while also serving as a point of interaction users and their device. On the other hand, all interactions that have to do with sending and receiving service requests are mediated by the ReL module. Also, the ReL communicates with the lightweight database to access hosted Web Services.

By design, the ReL is only active on a device that has turned on its service providing functionality, while the device can take advantage of energy-saving mechanisms like the Opportunistic Power Save protocol (OPS), and the Notice of Absence protocol (NOA) that are offered by the Wi-Fi Direct technology.

1.4 Discovery Engine (DiEn)

The core service discovery operation is driven by the DiEn through the Discover (DiM) and Ranking (RaM) Modules. A lightweight keyword-based Web Service matching algorithm suitable for this operation is prosed in [19]. This operation occurs in two phases: 1) Adaptation phase: this phase is triggered when the DiM receives a services request from the SRM. The DiM then obtains context information from the NoM and integrated them into services requests. This process essentially enriches the original service request with the device's context parameters to facilitate the discovery of relevant Web Services. 2) Ranking phase: where the discovered web services are then ranked according to their degree of suitability to the current context of the client device. Service requests from client devices are detected by the Service Request Listener module (ReL), prompting the host device to respond accordingly. Also, the SRL communicates with the LDb to access hosted services.

4. FUNCTIONAL DESCRIPTION OF USE CASES

From a functional perspective, the use case of the proposed architecture is summarized into two main tasks i) get device context and ii) get relevant services.

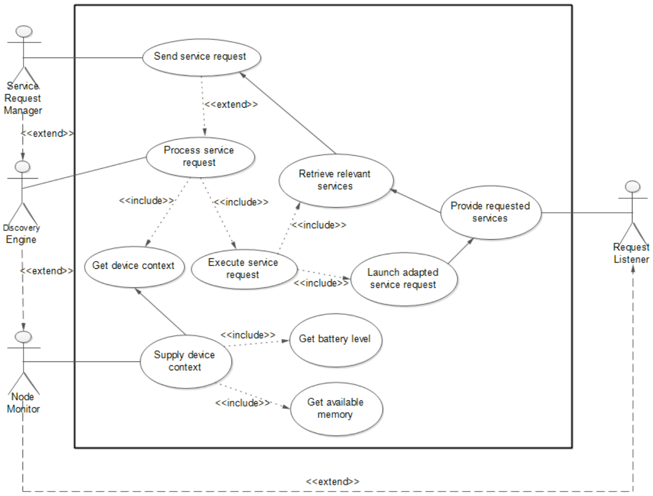

Figure 3: Use Case Diagram of Function Components

The first task results in the extraction of context information used for request adaptation and the second performs the actual service discovery and ranking process. And as shown in Figure 3, these tasks are handled by four use case actors: SRM actors, DiEn actor, NoM actor, and ReL actor

With the NoM actor, the "get device context" task is executed while the ReL, SRM, and DiEn actors collaborate to execute the 'get relevant services' task. A stepwise representation of how these tasks are handled by the respective actors is given in the activity diagram of Figure 4. As shown in the activity diagram, the DiEn actor is at the core of the chain of service discovery activities. It receives service requests, and then queries the NoM actor for context information. With this information, the DiEn actor performs service request adaptation before placing an actual request to the host device by engaging the ReL actor.

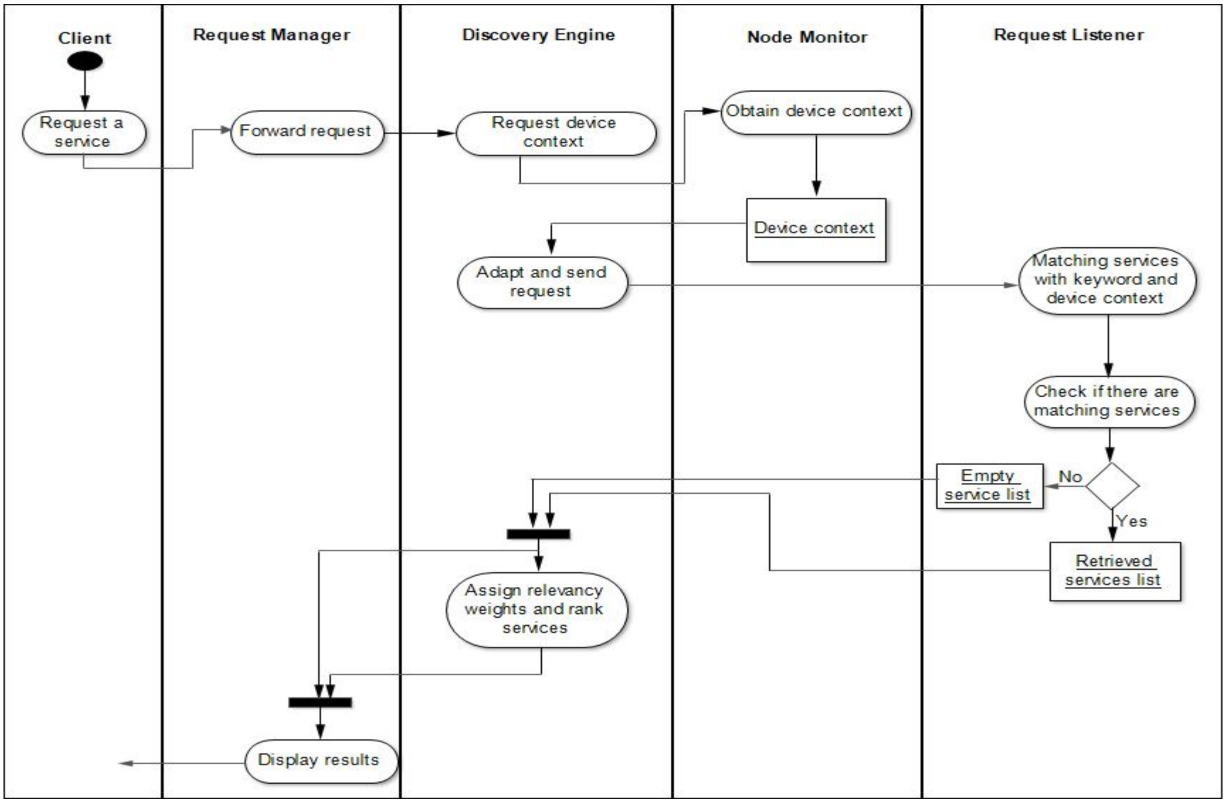

Figure 4: Use Case Activity Schedule

As depicted in Figure 4, the service discovery process that begins from when a service request is launched to when services are discovered and ranked ultimately invokes a chain of coordinated activities that are driven by the sequence of interaction between the different use case actors as illustrated Figure 5.

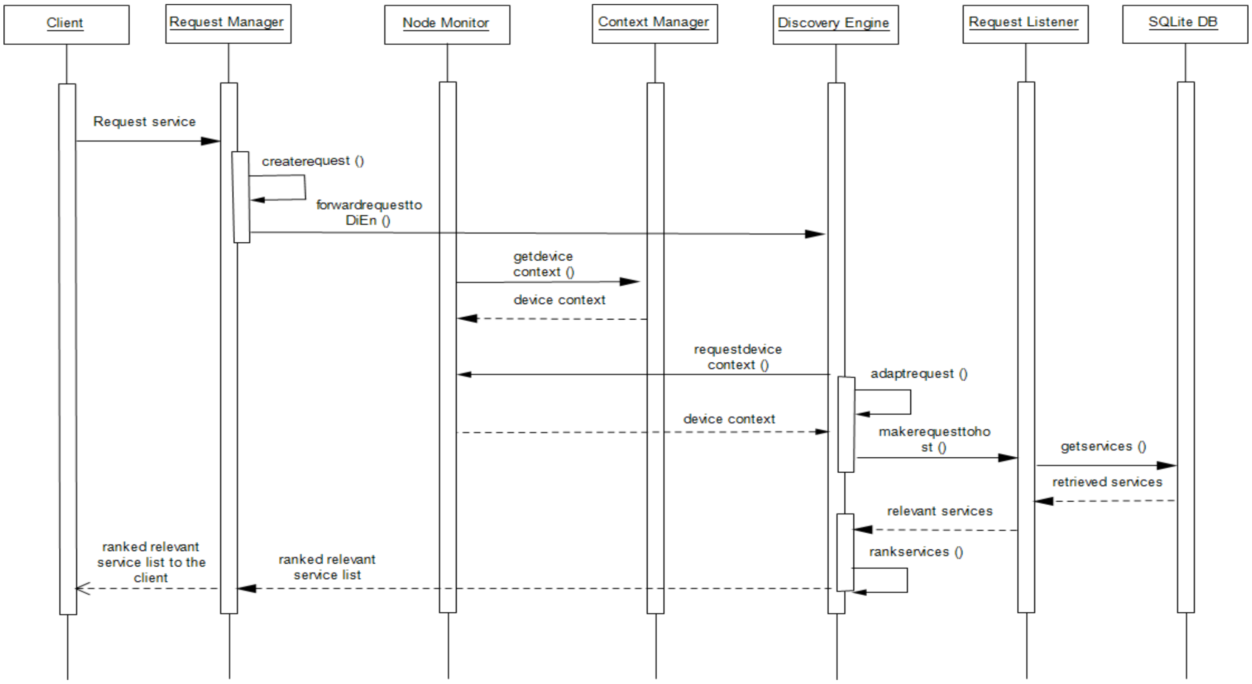

Figure 5: Interaction Between Use Case Actors

5. CONCLUSION

The emerging AMC paradigm advocates a mobile web service provisioning platform capable of complementing MCC. The idea is to enable mobile devices to collaborate in hosting web services in a manner that allows them to create an opportunistic and inexpensive pool of services that can be discovered and invoked among peer devices whenever the main cloud is inaccessible. The need for such a platform has become imperative because of the increasing global reliance on cloud services due to the widening acceptance of MCC and the explosive growth of IoT.

However, realizing service discovery in AMC is challenged by its unique dynamic and resource-constrained nature, which renders conventional MCC techniques unsuitable for use. As a result, lightweight service discovery mechanisms that prioritize the resource capabilities of the mobile device are required. This requirement primarily entails the design of appropriate architectures that can support resource-aware service discovery. This paper, therefore, contributes toward the realization of AMC in general by first advancing the "device-centric" approach to service personalization. The approach introduces the idea of using device resource resources and device preferences as key context information to determine service relevance in AMC. We then build on this idea to propose an architecture that supports resource-aware service discovery in AMC, presenting its technical design. To aid the actual implementation of the proposed architecture, a functional description of the architectural components was rendered in the form of UML diagrams.

REFERENCES

[1] M. Attaran and J. Woods, "Cloud computing technology: improving small business performance using the Internet," *J. Small Bus. Entrep.*, vol. 31, no. 6, pp. 495–519, Nov. 2019.
https://doi.org/10.1080/08276331.2018.1466850

[2] A. Yassine, S. Singh, M. S. Hossain, and G. Muhammad, "IoT big data analytics for smart homes with fog and cloud computing," *Futur. Gener. Comput. Syst.*, vol. 91, pp. 563–573, Feb. 2019.
https://doi.org/10.1016/j.future.2018.08.040

[3] C. Stergiou and K. Psannis, "Recent Advances Delivered in Mobile Cloud Computing's Security and Management Challenges," in *Modern principles, practices, and algorithms for cloud security*, Gupta, Bri., IGI Global, 2020, pp. 1–23.

[4] F. Jamal and R. Z. Khan, "Emerging Technologies and Developments in Cloud Computing: A Systematic Review," *Int. J. Emerg. Trends Eng. Res.*, vol. 8, no. 3, pp. 894–905, 2020.
https://doi.org/10.30534/ijeter/2020/46832020

[5] D. A. Egbe, B. M. Murimo, and M. O. Adigun, "A Device Resource-aware Service Discovery Solution for Ad-hoc Mobile Cloud," *Sci. Technol. Int. J. Sci. Technol.*, vol. 1010, no. 1, 2017.

[6] O. S. Neffati and O. Saidani, "Semantic Web Service Discovery Approaches : A Comparative Study," *Int. J. Comput. Sci. Mob. Comput.*, vol. 9, no. 2, pp. 87–95, 2020.

[7] M. Attaran and J. Woods, "Cloud computing technology: improving small business performance using the Internet," *J. Small Bus. Entrep.*, vol. 31, no. 6, pp. 495–519, 2019.

[8] S. H. Kim and J. K. Kim, "Determinants of the adoption of mobile cloud computing services: a principal-agent perspective," *Inf. Dev.*, vol. 34, no. 1, pp. 44–63, 2018.
https://doi.org/10.1177/0266666916673216

[9] M. Intelligenc, "Mobile Cloud Market Size, Share, Trends, Forecast (2020-25)," 2020. [Online]. Available: https://www.mordorintelligence.com/industry-reports/global-mobile-cloud-market-industry. [Accessed: 11-Jun-2020].

[10] E. Badidi, Y. Atif, Q. Z. Sheng, and M. Maheswaran, "On personalized cloud service provisioning for mobile users using adaptive and context-aware service composition," *Computing*, vol. 101, no. 4, pp. 291–318, Apr. 2019.

[11] S. Al_Janabi and N. Y. Hussein, "The Reality and Future of the Secure Mobile Cloud Computing (SMCC): Survey," in *Lecture Notes in Networks and Systems*, vol. 81, Springer, 2020, pp. 231–261.

[12] D. M. Simanjuntak and G. P. Kusuma, "Mobile Application Using Location-Based Service for Supporting Tourism Industry," *Int. J. Emerg. Trends Eng. Res.*, vol. 8, no. 4, pp. 1079–1085, 2020.
https://doi.org/10.30534/ijeter/2020/21842020

[13] Y. Perwej, F. Parwej, M. M. Mohamed Hassan, and N. Akhtar, "The Internet-of-Things (IoT) Security: A Technological Perspective and Review," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 5, no. 1, pp. 462–482, 2019.

[14] K. Samdanis and T. Taleb, "The Road beyond 5G: A Vision and Insight of the Key Technologies," *IEEE Netw.*, vol. 34, no. 2, pp. 135–141, Mar. 2020.

[15] B. Fekade, T. Maksymyuk, and M. Jo, "Clustering hypervisors to minimize failures in mobile cloud computing," *Wirel. Commun. Mob. Comput.*, vol. 16, no. 18, pp. 3455–3465, Dec. 2016.

[16] T. Verbelen, P. Simoens, F. De Turck, and B. Dhoedt, "Cloudlets: Bringing the cloud to the mobile user," in *MCS'12 - Proceedings of the 3rd ACM Workshop on Mobile Cloud Computing and Services*, 2012, pp. 29–35.
https://doi.org/10.1145/2307849.2307858

[17] Q. Wu and L. He, "Analysis of Social Media User Network Access Behavior Based on Web4.0," in *6th*

International Conference on Dependable Systems and Their Applications (DSA), 2020, pp. 127–134.

[18] C. Y. ; K. H. ; H. Chae, “Energy Efficient Mobile Cloud Computing Powered by Wireless Energy Transfer - IEEE Journals & Magazine,” *Sci. World J.*, vol. 34, no. 5, pp. 1757–1771, 2016.

[19] C. Stergiou, K. E. Psannis, B. G. Kim, and B. Gupta, “Secure integration of IoT and Cloud Computing,” *Futur. Gener. Comput. Syst.*, vol. 78, pp. 964–975, Jan. 2018.

[20] D. A. Egbe, A. O. Akingbesote, M. O. Adigun, and M. B. Mutanga, “Context based service discovery algorithm for ad hoc mobile cloud,” in *2016 International Conference on Industrial Informatics and Computer Systems (CIICS)*, 2016, pp. 1–6.

[21] D. A. Egbe, “Context-Aware Discovery of Relevant Web Services in Ad-hoc Mobile Cloud,” University of Zululand, South Africa, 2017.

[22] P. Hagen, H. Manning, R. S.-F. Research, U. Cambridge, U. MA, and U. 1999, “Smart personalization,” *Forrester Res. Cambridge, MA*, 1999.

[23] H. Wang, Z. Wang, S. Hu, X. Xu, S. Chen, and Z. Tu, “DUSKG: A fine-grained knowledge graph for effective personalized service recommendation,” *Futur. Gener. Comput. Syst.*, vol. 100, pp. 600–617, Nov. 2019.
https://doi.org/10.1016/j.future.2019.05.045

[24] N. Cheniki, Y. Sam, N. Messai, and A. Belkhir, “Context-aware and linked open data based service discovery,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018, vol. 10845 LNCS, pp. 448–462.

[25] K. Elgazzar, H. Hassanein, and P. Martin, “DaaS: Cloud-based Mobile Web service Discovery,” *Pervasive Mob. Comput.*, vol. 13, pp. 67–84, 2014.
https://doi.org/10.1016/j.pmcj.2013.10.015