TOWARDS TRULY SMART CITY SERVICE PROVIDERS: A VIEW ON ON-DEMAND EVERYTHING AS A SERVICES

Abdulsalam Alammari, Salman Abdul Moiz, Atul Negi

SCIS, University of Hyderabad SCIS, University of Hyderabad SCIS, University of Hyderabad
Hyderabad, India Hyderabad, India Hyderabad, India
alammari@uohyd.ac.in
salman@uohyd.ac.in
atulcs@uohyd.ac.in

Received: 08.02.2020 Revised: 16.03.2020 Accepted: 13.04.2020

Abstract
Internet of Things (IoT) connects a large number of devices/objects, and these devices are being increased that it is foretold their number will reach billions in the near future, and hence increasing the number of IoT-based services. IoT is built on and integrated with many existing technologies (e.g., wireless sensors, mobile computing, and cloud) which support its spreading and scalability and help it to overcome its known challenges like limited resources. Moreover, service composition supports IoT in the rapid development of new IoT applications leveraging the existing services. These paradigms and techniques are fundamental building blocks and enablers of smart cities. IoT service composition and IoT integration with the above-stated technologies have been addressed in research and industry works. In this paper, we review several types compositions and give our view on future smart city service providers. We also propose a location-aware architecture for integrating IoT services and mobile application services. The proposal is illustrated based on shopping mall scenario, and the high-level architecture is presented in this work.

Keywords—Internet of Things, Cloud, Mobile Applications, Service Composition, Service Providers, Smart City, Sensing as a Service, Actuating as a Service, Everything as a Service.

INTRODUCTION
IoT was defined in [1] as Interconnection of sensing and actuating devices providing the ability to share information across platforms through a unified framework, developing a common operating picture for enabling innovative applications. This is achieved by seamless ubiquitous sensing, data analytics and information representation with Cloud computing as the unifying framework.

IoT refers to a new a new range of connecting devices/objects (other than the known networks of computers) watch, curtain, juicer, TV, car, kitchen appliances, health devices and much more. As the Internet is spreading and the microchips and sensors/actuators are getting cheaper, more Internet of Things applications come to the reality leading to more types and number of connected devices. IoT aims at connecting people, places, data, and things anywhere anytime to create a smart world based on smart wearable/room/home/building/parking, wider to smart sensing and Actuating networks, transport, smart grid, and reaching smart cities. Hence, IoT enables such physical objects to communicate, coordinate and share their data by employing the pre-existing technologies. This arrangement is more suitable for these physical objects and allows them to cope with their constraints and limitations to make them smart, rather than as traditional passive objects [2]. In other words, IoT is the enabler connecting the physical devices to the digital world, making numerous physical devices to be connected to the internet and allow remote control and heterogeneous devices collaboration fig.1.

![Internet of Things Concept](image-url)

Fig. 1: An overall picture of Internet of Things concepts
IoT devices have well known limitations and constraints [e.g., limited storage, limited processing and communication capabilities], thanks to cloud paradigm that serves as a backbone to help IoT overcome such limitations by providing remote processing and storage of IoT devices data. Cloud provides IoT with virtually unlimited resources [5]. Besides, sensors as essential components of IoT applications create a large amount of data (Big Data). Big Data has its associated challenges like storage and communication that IoT constrained devices cannot address without the great solutions provided by IoT integration with Cloud. This integration supports Storage as a Service [6]. Software as a Service and Platform as a Service [7]. However, the more sensors are connected, more data is sent to cloud leading to increasing transmission overhead and latency issues [8].

[9] Defined IoT services as services that enable interaction with the physical world and emphasized that IoT services can be stand-alone as a field on their own. Compared to Internet services, which run on powerful and rich resources (power and computation ability) computers, IoT services are deployed and run on resource-constrained devices. Besides, and due to the unreliability of connection of wireless links, IoT services might be present or missing once needed. Hence the need raised for IoT suitable plans for choosing and composing services [10].

IoT and pervasive computing are supposed to provide services to the user to meet their requirement and with minimal or no user intervention. In some cases, the requirement of users may be fulfilled with an atomic service. However, in IoT a lot of services and services sources (usually resource-constrained devices with digital services) are available, but one key challenge is allowing easy and dynamically composition of available distributed and heterogeneous services to collaborate to perform an advance or complex task which may not be fulfilled using one atomic service. Later, the composite service may be used to fulfill other users requirements in different scenarios [11].

The fact that IoT consists of a massive amount of heterogeneous devices that may freely join and leave the network, this face rise another challenge- during runtime the device may be missing, and hence the required service may be missed [1]. This challenge increases the need for dynamic runtime service composition which of course required automatic discovery and selection.

The domain area of mobile computing and Internet of things was Defined in [12] as the research area that involves case studies, prototypes, demonstrations, applications and business cases of the IoT/WoT, through mobile phones, where the user of the mobile phone interacts with cyber-physical things that are enabled to the Internet/Web, through his/her phone device, exploiting at the same time the sensing capabilities of the phone.

Mobile devices like smartphone, tablets, and wearable alongside with their application strongly present in a smart room, smart home, smart transport, and smart city and even more. These devices and their applications are the most important interfaces through which the user interacts with IoT devices why not since they have many connectivity means like Cellular, Bluetooth, NFC (Near Field Communication), and WiFi by which they can communicate with cloud, sensors/actuators and among themselves. GPS (Global Positioning System) is also one of the sensors available in the mobile phone that are useful for location-based IoT applications which are many, and indeed location is an important parameter for IoT applications particularly sensing and actuating based once. Besides, the decreasing price of smartphones with the mentioned features makes them the main and the first choice as IoT interface.

Mobile applications have gone through stages in the scope and targeted users they serve. First, they were targeted to users in their mobile phones then were widely adopted by enterprise and business. Currently, mobile applications play an important role in managing IoT devices and smart work [13].

Besides Cloud and Wireless, no doubt the maturity of mobile applications industry has a noticeable contribution to the widespread of many technologies, including IoT, which changes the way people interact business and luxury lifestyle especially nowadays as people use their mobile phones for majority of their daily activities. We can call here, to name few, some successful business like UBER and OLA cab companies that mainly depend on mobile applications to provide their services with high level of customers trust.

Besides its role in enabling scalability, service composition also supports the fast development of new and creative IoT applications by leveraging the currently available services and composition models.

Existing approaches of IoT application development are usually polarized either towards responsiveness or flexibility. Most approaches rely on precompiled service mashups created manually by developers and end-users, which results in IoT applications that are highly responsive, but inflexible due to their static nature. At the other end of the spectrum, fully automatic composition of services at runtime results in IoT applications that are highly flexible, but the service composition overhead grows exponentially with the number of IoT services in the environment.

To emphasize both application flexibility and responsiveness, we developed a location aware approach that relies on the decentralized resources of virtualized services for goal-driven composition of precompiled service mashups: location-aware software agents are equipped with libraries of plans that they use to achieve their goals, and cooperate with one another to compose IoT mashups at runtime in pursuit of their goals.

Evaluation results show that, in the worst-case scenario, the mashup composition overhead grows only linearly with the number of agents, which would typically be less than or equal to the number of devices in the environment.

An architecture for integrating IoT services and Mobile applications is proposed in this paper with discussion on its components based on an example scenario. A view of future smart city service providers is also discussed.

The rest of this paper is organized as follows. A study of foundational concepts of IoT-based applications and related work is presented in section 2. In section 3, we explain our proposed architecture for integration, an example scenario, its architectural design, and Evaluation. We conclude the paper and outline the future vision in section 4.

FOUNDATIONAL CONCEPTS AND RELATED WORK

We start this section with a brief discussion on IoT main building blocks and fundamental concepts required by any IoT-based application to achieve its functionality. This is followed by review of related work.

Foundational Concepts
- Identification: There are billions of IoT devices and services; hence IoT needs to identify them uniquely. There is a difference between object ID (that refer to the object name or the services name like wind speed sensor or D1) and its unique address in the communication network [2].
- Sensing: Sensing means reading the environment data (observations) using the sensors devices and upload it for analysing or storage [2].
- Communication: There are several heterogeneous and resource constrains IoT devices in which they require low power for operation hence require light weight communication protocols that support such needs. Some of the communication protocols used in IoT are, Bluetooth, WiFi, IEEE802.15.4, LTE-Advanced, Z-wave, RFID (Radio Frequency Identifier), Ultra-Wide...
Bandwidth (UWB) and Near Field Communication (NFC) [2].

- Computation: IoT applications require processing units (hardware and software) to run and provide the computational mean. Arduino, Raspberry Pi, WiSense, Cubeieboard, Z1, BeagleBone, Intel Galileo, FriendlyARM are some examples of hardware platforms commonly used for running IoT applications. (RTOS) Real-Time Operating Systems like Contiki RTOS with its simulator - Cooj- is used to devise real-time IoT-based applications. TinyOS, LiteOS and Riot OS are also lightweight operating systems designed to suit for IoT environments [2].

- IoT Services: Are services that enable interaction with the physical world. IoT Services can be stand-alone as a field on their own. Compared to Internet services, which run on powerful and rich resources (power and computation ability) computers, IoT Services are deployed and run on resource-constrained devices [9].

- Semantics: Refers to the ability of IoT heterogeneous devices to extract knowledge smartly out of the sensed raw data. This takes great support of Semantic Web technologies like Resource Description Framework (RDF) and the Web Ontology Language (OWL) that help in describing and representing information in machine-understandable ways [2].

- Cloud Computing: It is the provision of providing several types of services and resources through the internet. These services are like platform, software, computation, storage. It has become a common choice for individuals and enterprises as it saves development costs and increases security and performance [14].

- Infrastructure as a Service (IaaS): Is one form of cloud computing where infrastructure components like (servers, networking hardware, storage) are being hosted at the cloud of the service provider and mad accessible to user over the internet [14].

- Platform as a service (PaaS): Is the form of cloud computing where the service provider offers hardware and software tools and make them accessible over the internet [14].

- Software as a Service (SaaS): Is the provision that allows users (individuals or enterprises) to connect to and use cloud-based applications over the Internet. Here the application are hosted at the third party service provider and mad accessible for user through the internet [14].

- Fog Computing: It is the paradigm that brings some of the cloud capabilities (computing, storage and applications) closer to the data sources. The location of nodes is not necessarily fixed; they maybe somewhere between the data sources and cloud [15].

- Sensing as a Service (SaaS): Service providers introduce observations of particular sensors to end-users as a service free of cost or on rent basis. The sensors may belong to the service provides themselves or being rented by devices owners [16,17].

- Sensing and Acting as a Service (SaaS): It is quite similar to SaaS adding to it that providers also provide applying forces to particular actuators as per end-users requirements [18].

Related Work
Following we review some research studies and surveys in the integration of IoT and other technologies paradigms like cloud, wireless sensor networks, and mobile phones. Also, several works in using stand-alone sensors/actuators, mobile phone integrated sensors/actuators for many sensing and actuating applications. The main focus here is to show the types of integrations and compositions of several technologies and paradigms that resulted in a new view of sensing and actuating applications and architectural views. To name few, mobile phones and mobile computing, cloud computing, edge and fog computing.

[19] Developed a location-aware mobile application to locate and interact with web-enabled sensors to provide the user with the local environmental condition.

To support continuous patient roaming within or outside the hospital [20] introduced healthcare monitoring system which includes a simple analysis at the cell phone. It was designed with two ways of transmitting data to the server: in case the body sensors are within the LAN coverage area, then LAN internet will be used to transmit the data. If the sensors are outside the LAN coverage area, then the sensed data will be gathered directly to the phone which will transmit them to the server using CDMA network.

Park here smart parking system in [21] leverages the embedded sensors and short-range communication in mobile phones for finding and disseminating the parking availability. By analyzing sensors state (accelerometer, gyroscope) and by identifying the change of mode of transportation (from car to walking and from walking to car), the parking actions can be identified automatically. Two modes of parking spot data dissemination: local, device-to-device, using WIFI Direct technology to disseminate data around the user locality. Global, to a remote server, via a 3G/4G network. It worth mentioning here that due to the absence of sensors infrastructure in the parking and the fact the Park Here does not use GPS, Park here smart parking system is a per-cell probability to find a vacant parking slot hence it does not geolocate the specific slot.

AndWellness system "personal data collection system" [22] uses mobile phones to collects data form participants to conduct surveys and display the participant’s statistics in real-time for both participants as well as researchers. The participants are prompted in their mobile at a configured time intervals to answer surveys (time-based, location-based, or other contextual parameters) where the answers are sent wirelessly to the server. Besides the configured intervals, the participant’s mobile sensors can be used to upload continuous data with user location tracking using GPS. SenSay (a single sensor-based approach) [23] was developed to identify the particular mobile device context using only the embedded sensors in that device. It realises the change of context of the mobile user (standing, sitting, walking or running), In SenSay, there is no cooperation between sensors belongs to different handsets. Alternatively to the use of a single integrated sensor, [24,25] used only integrated sensors in the handset, but here multiple simple sensors are integrated to identify the context cooperatively.

In SPA [26] - mobile-based system - a collaboration between body area sensor network and environment sensors is suggested to continuously sense and upload the gathered data to a remote server to be analysed by professional for continuous health monitoring.

CONSORTS-S [27] designed a platform to provide mobile users with context-aware services by accessing wireless sensor networks in the surrounding. CONSORTS-S architecture consists of three main parts; 1) The mobile phone to be used as an interface. 2) For overcoming the mobile phone limitations, CONSORTS-S platform has attached mobile sensor router to the mobile device to help in the communication with the surrounding wireless sensors. 3) Middleware located in the remote server for management and sensed data analysis.

Urban Sensing was addressed in several studies. Some of such works were designed as application-specific. CodeBlue [28] introduced a wireless infrastructure for emergency medical care application. CodeBlue integrates low-power, wireless vital sign sensors, Personal Digital Assistants, and PC-class systems with that results in a network of dense devices. Ako, [29] developed a general architecture for industrial application to monitor and predict equipment failure.
UrbanRadar [19] - location-based application - an application which discovers and communicates with environmental services - provided by Web-enabled urban sensors. In UrbanRadar, user can create urban mashups, which are satisfied only when several predefined conditions (from the mashup services) are met. As an example, a location-aware environment monitoring approach introduced in UrbanRadar to make users updated with the current real-time environmental conditions. Their novelty is that they reduce the direct citizen involvement by reading the data through his mobile device and share them over the web as a contribution towards real-time digital city vision supporting urban sensing. Their developed mobile application interacts with web-enabled sensors deployed in the environment around the user. Finally, a general purpose framework, AnonySense [30], to allow participatory sensing to answer the application’s query about the context leveraging the sensors on participants mobile phones taking the privacy of participants into account.

People-Centric UrbanSensing [31] was an attempt to encourage the movement of sensors network from application-specific – mostly small scale- to the scalable and general purpose network for commercial mainstream which can support a variety of urban application (buildings, cities, enterprises). Moreover, People-CentricUrbanSensing is designed to extend the general purpose infrastructure to “new sensing edge for the internet.” People-CentricUrbanSensing architecture favours the opportunistic sensing-based approach to support its scalability. Out of several main sensing elements that urban sensing is comprised of - people, buildings, vehicles - People-CentricUrbanSensing follow a people-centric approach. Several issues like (the coordination among static sensors, people-centric mobile sensors and edge wireless access nodes) that aim opportunistic sensing, tasking and data collection. Besides, challenges like wide coverage of sensing areas with numerous mobile sensors, responsibility hand of sensors coordination, network performance, privacy and security were also discussed in this work for the support of the new view. MetroSense [32] also based on people-centric sensing approach that follows three main stages (sense, learn and share). MetroSense sensing is based on three sources of sensing: mobile phones sensors, fixed sensors in civic infrastructure, and edge wireless access node which provide the internet gateway.

Eco-feedback sensing systems was developed with mobile phone interface to enable monitoring the power consumption [33] with the use of a mobile camera (pointing it to the device)[34].

Participatory sensing for large scale sensing (the evolution at mobile crowd sensing using internet-enabled smart phones) as an enabler for a broader range of urban applications like traffic planning, environment monitoring and public safety [35,36].

Using smartphones as a source of sensing services was first introduced and termed as Sensing as a Service (SaaS) in [16], and further elaborated and detailed by the same authors in [37]. SaaS concept strongly depends on cloud computing as an essential computing model with many services offered which can be leveraged by this new concept. The work also detailed some fundamental requirements for any sensing as a service via smart phones. 1) Ability to support different mobile platforms serving several types of mobile phones sensing applications. 2) Having in mind that such devices are power constraint, it is also required that the system is power efficient. 3) Support for efficient interconnection mechanisms. Sensing request can be handled from different locations with the help of the cloud and the deployment of sensing server in different locations. Besides, [16] stated the following functionalities to be supported by SaaS Cloud. Web interface to be provided for users accesses both from mobile devices and normal computers. Ability to generate new sensing task (standarized format including what data to be collected, sensors to be used and area of interest) for each new data collection request.

Ability to recruit participants to participate in sensing for each new request. The ability to schedule the sensing activities between mobile participants. Ability to deploy sensors managing applications on the participant mobiles such that the data sensing and sending them to the server are managed and under control. Finally, the ability to save data for future use.

Spheres [38]: A web services framework for smartphone sensing as a service. Spheres introduced (SOA) services oriented architecture based framework integrating mobile internet, wireless sensor network and cloud to provide smartphone crowd sensing to be shared for both public and private use. Spheres is based in the author previous work [39] with the main difference is that mobile phone sensors are not in fixed location comparing the [39] where wireless sensor nodes were connected to a gateway and formed wireless sensor network.

In SaaS [40], the word sensors refers to physical sensors as well as virtual sensors. Hence, any sources that generate data, social media accounts and weather APIs, for example, are considered virtual sensors.

Mobile phones, as well as standalone sensors/actuators, are being used to actuate and control physical devices in several domains, and a number of models and architectures have been suggested in the literature for the same either for specific domain applications or generally proposed.

Internet technology along with Internet of Things are foreseen to be the future standards, particularly for home automation and control [41,42], providing interoperability between heterogeneous types of devices. [43] Web protocols used to connect smart meters to mobile phones interface to provide users with real-time device level feedback that allow users to monitor and control energy consumption in their homes. Also [44] provided general mobile phone application that provide the ability of creating physical mashup of composition of different smart things.

IoT and Cloud both mutually benefit from each other. To overcome its essential constraints (e.g., limited storage, limited processing and communication capabilities) Cloud provide IoT with virtually unlimited resources [5]. Besides, sensors as essential components of IoT applications create a huge amount of data (Big Data). Big Data has its associated challenges like storage and communication that IoT, with its constraint resources, cannot address without the great solutions provided by IoT integration with Cloud. This integration by Storage as a Service [45], Software as a Service and Platform as a Service [7]. Cloud also support the development of scalable IoT applications as proposed in [46] open source IoT Cloud framework and its extensible API.

Cloud, on its turn, benefits from IoT as an enabler to deal with real world things giving Cloud means for creating numerous services in a distributed manner. Many IoT and Cloud-integrated applications are built around the above discussed concepts with Cloud being used as backend and IoT as frontend.

Sensing Cloud Infrastructure SCI was introduced in [47] to provide virtualized (on the cloud) set of physical sensors required by the user on demand. User has control and monitor over the virtual set of sensors on IT infrastructure and out to destroy them once they are not in need. SCI contains five servers, mainly, portal server, service provider server, resource management server, monitoring server and virtual server. SCI was extended to the concept of Internet of Things in [48] which comprises three layers: client-centric, middleware, and sensor-centric. SCI was optimized in SenseCloud [49], a cloud based solution with unified access to different sensor networks and a level of abstraction that hides the underlying complexity. Sensing framework (sensing cloud) involves a sensor virtualization mechanism, a multitencency mechanism, and a dynamic provisioning mechanism.
Authors in [50] surveyed the use of cloud computing as to provide remote sensing services (Cloud-assisted remote sensing, CARs for short) as an enabler of Internet of Everything (IoE). Thus, they believe it leads to prompting smart cloud services. Through real-world application, researchers explained the benefit of using cloud-based services to empower remote sensing. Besides, a four-layered architecture (CARs) is proposed in this work: 1) fog layer; 2) stratus layer; 3) alto-cumulus layer; and 4) cirrus layer. CARs mainly provide three service models: Sensing and Acting as a Service (SaaS), Sensing and Acting Platform as a Service (SAPaaS), and Sensing as a Service (SDAaaS), which are alike to cloud computing service models: IaaS; PaaS; and SaaS. It worth mentioning here that, in CARs architecture, sensing and actuating resources are encapsulated in the fog layer. Cloud of Things (CoT) [51] introduced cloud architecture for sensing as a service. Authors hypothesis that taking benefit of Internet of Things devices global sensing resources in cloud platform to sport remote sensing is an efficacious way to achieve their sensing as a service foresight in their previous work, CARs [50]. In contrast to the traditional cloud platforms that collect data for later use, Cloud of Things processes the data in-network by IoT devices to directly provide meaningful information.

Sensing and Acting as a Service (SaaS) provides various types of services and/or actuators (S/A) from resources from sensor networks and personal mobile devices to the end user as services. The end user can then build their own application based on rented composed/aggregated types of resources of their desired locations, functionalities and time. SaaS also provides the end user with the ability to trigger actuating commands as per their analysis of the scenarios/real/historical data etc.

In SaaS owners (being individuals, enterprise, universities, etc.) of sensing/actuating devices may contribute their devices services for free or provides the same on rent bases. Devices owners may contribute standalone Sensors/Actuators (S/A) or WSIN in which they are called contributing nodes which here (whether have single or group of sensors or actuators) must belong to the same administrative domain [18]. End users (the ones will take the benefits of the sensors and actuators) might also be individuals, enterprises, universities, etc.

The SaaS provider acts as the mediator between the two parties and manages all the aspects of the paradigm. Among SaaS provider responsibilities are devices owners and end user enrollment, Service Level Agreement creation, physical nodes selection, rent calculation, virtualization, etc. SaaS virtualization has two types. First is the virtualization of the observation data in which the provider receives the sensors observations and saves them to provide each end user with the required data at their prespecified time. Second is the actuator virtualization where the provider facilitates the end user access (sending an actuator command) to the previously rented actuator. This concludes that the provider must create a virtual set of desired sensors and actuator for each end used as per their requirement.

Authors of [18] identified the main concepts and actors of SaaS and proposed an architecture of SaaS in which they make a hypervisor in the nodes to abstract the the creation, management and virtualization of sensors and actuators as a virtual instances in the cloud. Even though the hypervisor works at device level (device level virtualization and abstraction), it enables direct communication with sensing and actuating devices through the Automatic Enforcer which is located between the VolunteerCloud Manager and the Hypervisor. In [52], authors used the hypervisor proposed in [18] to add more detailed architecture for SaaS giving more concentration for management, abstraction and virtualization of sensing resources. Their proposed architecture allows devices to provide their sensors as virtual instance which makes a particular physical device handles concurrent request. Besides, the architecture can compose/aggregate a network of resource instances out of simple individual instances. It worth mentioning here, since the hypervisor is the base of this architecture, that this architecture is device-oriented approach but still concurrent requests of the same physical device are ensured. Further extension to their works in [18] and [52], authors proposed utility framework [53] for IoT SaaS approach inside IoT-A reference architecture[54]. This work implemented the earlier proposed idea (implementation for Android mobile) in a real life IoT scenario to show its feasibility.

Stack4Things [55] adopted the OpenStack [56] (Infrastructure as a Service framework) to propose a framework for Sensing and Acting as a Service (device-centric approach). Stack4Things shows the detailed subsystems for resources management and their observation data, and some use cases were demonstrated.

Internet of Things Sensors and Actuators Layered Fog Service Delivery Model (SALFSD) [57], authors proposed a service delivery model for sensing and actuating as a service delivery model. From bottom-up, SALFSD consists of three main layers: Sensing Layer; sensors and actuators devices connected to dumb gateways (gateways only forward the observation to the upper layer and forward commands to actuators, no processing at gateways). Layered fog, which consists of several layers of fog nodes to monitor and manage the sensing layer and coordinate with the upper layer. Cloud layer, responsible for managing all layers, users requests, virtualization, accounting, etc. Layered fog in SALFSD provides fault tolerance and fasten the monitoring and taking the decision related to the connected gateways in the lower layers.

**Discussion about Related Work**

In this section, we have presented the foundation concepts and several pre-existing technologies and paradigms on which sensing services and Internet of Things are built upon and depend on for communication, architecture, storage, sensing, arrangement and more. Then we introduce some related work that used such technologies to cover IoT limitation and increase its acceptance for building IoT-based applications.

The related work covered several ways of sensing services; using fixed sensors, mobile phone sensing, echo feedback sensing, urban sensing, people-centric sensing, community sensing, participatory sensing, opportunistic sensing, and sensing cloud. The flow of works presented this section shows opportunistic sensing provides more scalability and reduced human involvement in sensing application. Such scalability got more support and benefits as cloud becomes part of such paradigms. More opportunities and innovative applications emerged by using IoT and introducing sensing as a service and sensing and actuating as a service. Nevertheless, we noticed the lack of a single platform that integrates or bring together mobile applications services -software services provided by applications installed the mobile devices- and IoT services as all the researches addressed the use sensors and actuators only in the mobile devices.

Hence, we propose the aimed platform as a different view of integration; on-the-go integration of mobile applications and IoT services as illustrated in the following section. The discussion about its components is based on an example scenario.

**PROPOSED ARCHITECTURE**

Mobile applications industry has a good experience developed through the last few years. We here mention -for the sake of the example scenario- the commercial ones. Moreover, the recent emergence of IoT has gained wide customers’ acceptance, governments, and businesses. Together they make beautiful technologies working for more lifestyle applications and provide a lot of services from which we can select to compose or advice to the user.
In this work, we aim at making an intermediate architecture between IoT services in the locality and mobile applications services such that the integrating happens temporally on-the-go, as a form of advice-search for what can be useful at this time in this location being IoT services that user can access and use or mobile application services or coupon/deal etc. and give advice to the user.

We adopt the concept of location-aware recommenders in getting mobile apps services in the locality without following any specific scheme like publish/subscribe as we will list all available mobile applications services in the locality besides the IoT based services in the locality.

Some context parameters like location, date and time, will be taken into consideration to improve the accuracy of list of advised services.

Example Scenario: Shopping Mall
For the sake of simplicity, we have selected the Shopping Mall scenario as an example to illustrate our proposal throughout this paper. In this scenario, a user Sam enters into the shopping mall, upon arriving Sam’s location will be determined using his smartphone GPS. A search will be conducted to identify IoT services in the mall like parking navigation system (or availability of kids tracking devices) etc. And also a search in the commercial applications (like the popular applications in India; Paytm, Little, and nearby,PhonePe) for any deals to be used in shops, salons, cafe, or restaurants, etc. at this mall. Sam will get a list of all services from which he may use or buy a coupon to grab a deal. This may attract the customer to use the services as they are in physical proximity.

There are many other scenarios in which our idea of composing IoT with mobile applications services can be implemented. However, in this work, we selected a basic scenario for the sake of simplicity of delivering the idea.

Architectural Design
We hereafter explain the architecture based on the example scenario. The high-level architecture of our proposal is shown in fig. 2. And the operational procedure is shown in fig.3.
Fig. 3 The operational procedure of the proposed architecture

**Evaluation**

The evaluation mainly represents that the proposed architecture is more responsive than the existing architecture and also concluded that the proposed high level architecture was more flexible. IoT mashups are majorly composed through independent interactions between the agents. In the worst-case scenario, an agent would have to interact with all the remaining agents in the different locations to fulfill its purpose. We thus expect that, in this worst-case scenario, the mashup composition overhead grows linearly with the number of agents present in the various locations. The first objective of our responsiveness evaluation is to confirm this assumption. Our second objective is to estimate the number of sequential interactions that can be performed in a typical number of locations in less than 1 second. We consider 1 second to be a reasonable mashup composition latency for IoT applications that would classify as responsive in most use-case scenarios.

**CONCLUSION**

Supported by cloud features, IoT service providers - particularly sensing and actuating - are advised to be enhanced in such a way to provide service delivery models which compose types of services from different application domains. That eventually will make cloud service provider be viewed as an integration of several services provides. This is possible with the state-of-the-art accounting paradigms like pay-as-you-use that can govern the relationship among such several service providers. This view is more suitable for fast growth of IoT applications and new start-ups which don't have to invest for their own infrastructure as there will be many competitors service providers offering A to Z hardware and software requirements. This is a vision towards ready made, on demand, and heterogeneous smart city infrastructure which encourages fast and economically sound innovative applications. An architecture for integrating IoT services with mobile applications services is proposed in this work. The architecture makes the integration as a form of suggestion/recommendation of available and attracted services without the involvement of real composing to create new composite services. This idea is not restricted to shopping mall and can also be done in many domains as long as IoT and mobile applications services are available and may be useful at the same place, and indeed the type of services and mobile applications may differ according to the domain.

**REFERENCES**


56. OpenStack, https://docs.openstack.org/stein