MARGOT: Dynamic IoT Resource Discovery for HADR Environments

Lorenzo Campioni[1], Rita Lenzi[2], Filippo Poltronieri[1], Manas Pradhan[3], Mauro Tortonesi[1], Cesare Stefanelli[1], Niranjan Suri[4,2]

[1] Distributed Systems Research Group, University of Ferrara, Ferrara, Italy
{lorenzo.campioni, filippo.poltronieri, cesare.stefanelli, mauro.tortonesi}@unife.it

[2] Florida Institute for Human and Machine Cognition (IHMC), Pensacola, FL, USA
{rlenzi, nsuri}@ihmc.us

[3] Information Technology for Command and Control, Fraunhofer Institute for Communication, Wachtberg, Germany
manas.pradhan@fkie.fraunhofer.de

[4] US Army Research Laboratory (ARL), Adelphi, MD, USA
niranjan.suri.civ@mail.mil

***Abstract*—Smart City services leverage sophisticated IT architectures whose assets are deployed in dynamic and heterogeneous computing and communication scenarios. Those services are particularly interesting for Humanitarian Assistance and Disaster Relief (HADR) operations in urban environments, which could improve Situation Awareness by exploiting the Smart City IT infrastructure. To this end, an enabling requirement is the discovery of the available Internet-of-Things (IoT) resources, including sensors, actuators, services, and computing resources, based on a variety of criteria, such as geographical location, proximity, type of device, type of capability, coverage, resource availability, and communication topology / quality of network links. To date, no single standard has emerged that has been widely adopted to solve the discovery challenge. Instead, a variety of different standards have been proposed and cities have either adopted one that is convenient or reinvented a new standard just for themselves. Therefore, enabling discovery across different standards and administrative domains is a fundamental requirement to enable HADR operations in Smart Cities. To address these challenges, we developed MARGOT (Multi-domain Asynchronous Gateway Of Things), a comprehensive solution for resource discovery in Smart City environments that implements a distributed and federated architecture and supports a wide range of discovery protocols.**

***Index Terms*—Internet-of-Things (IoT), Smart City, Federated Resource Discovery, HADR.**

I. Introduction

The concept of Internet-of-Things (IoT) adoption in everyday life has gained momentum in the past few years [1], [2]. Processors, controllers, sensors, and actuators are getting embedded in everyday things impacting the interaction and perception of humans with their surroundings. Ranging from mobile devices, smart home assistants, and building sensors to industrial automation enablers, the scale and diversity of implementations of IoT-technologies has multiplied manyfold. Apart from the hardware aspects, there has been development in IoT related communication technologies, data communication protocols, security and privacy mechanisms, and interoperability aspects between IoT technologies as well as between IoT and legacy technologies [3]–[6].

Smart Cities are one environment that IoT applications have focused on [7], [8]. The migration of populations around the world to city-spheres has exerted significant pressure on city administrations to come up with new ideas to tackle the needs of the people, thus calling for the adoption of IT services and IoT-based solutions to improve the citizens' quality of life [9]. Smart city services are implemented on top of sophisticated IT architectures whose components are deployed in dynamic and heterogeneous computing and communication scenarios [10].

Even if the primary role of Smart City IoT capabilities is to provide valuable services to its citizens, they also present valuable assets for Humanitarian Assistance and Disaster Relief (HADR) operations, in which military, police forces and other local enforcement agencies could leverage the existing IoT infrastructure to coordinate and perform these operations [11], [12]. Furthermore, these operations may require HADR operators to integrate purpose-specific Internet of Battlefield Things (IoBT) applications that need to discover and interact with the existing IoT infrastructure [13], [14].

As a result, application components need to discover existing IoT and IoBT resources, mostly in their proximity but also in other geographical locations affected or participating in the HADR operations. Academia and the industry have dedicated numerous efforts to IoT asset discovery. However, most of the solutions have focused on the standardization of communication and discovery protocols or on the implementation of solutions for global/multidomain discovery on top of standard protocols [15], [16]. These solutions represent important milestones, but do not consider two fundamental requirements of large scale IoT and IoBT applications, namely: the need to support heterogeneous communication and discovery protocols and allow for fine grained control of information sharing about IoT assets across IoT domains.

To address these issues, we developed MARGOT (Multi-domain Asynchronous Gateway Of Things), a comprehensive solution for resource discovery in Smart City environments, which was specifically built to address the needs of both civilian IoT applications and IoBT applications. In particular, MARGOT implements location- and context-aware discovery of a wide range of IoT assets, including sensors and edge devices, e.g. IoT gateways for allocation of information processing tasks. MARGOT organizes IoT asset information in a distributed and federated database, according to the IoT domain in which the asset is located. Each IoT domain is associated with a MARGOT gateway to discover local IoT assets using a multi-paradigm and multi-protocol approach. Finally, MARGOT gateways implement fine grained information management policies (both at the application and IoT domain levels) that control the discovery of assets as well as the forwarding of queries between the different IoT domains.

II. A Scenario for HADR Operations

Smart Cities are increasingly being pervasively instrumented with a variety of sensors such as cameras, traffic, transportation, pollution / air quality, weather, noise, power, water, etc. Furthermore, individuals in these environments also interact with a variety of smart devices, such as smart homes, smart buildings, smart phones, health monitoring devices, etc. Smart Cities combine all of these sensors, actuators, and edge computing resources in interesting and sophisticated ways to enhance urban services for day to day activities and enable the development of innovative applications that improve the overall quality of life.

In these environments the role of Information and Communications Technology (ICT) is to provide to stakeholders a comprehensive view of the available assets as well as effective services that allow and simplify the interaction with such resources. To achieve this goal an ICT solution must consider and address the various properties and challenges that characterize these scenarios. In fact, assets are heterogeneous in nature both at the hardware and software level. They might adopt different communication protocols and standards, may be limited by their power constraints, ownership, and so on.

Specifically, IoT resources can be summarized and described by the information related to their context and their specific use. The type of a resource (e.g. sensor or actuator), its location or even the communication protocol adopted can be used to recognize assets' taxonomies in the network, which help stakeholders to effectively discover resources by using more sophisticated or specific search strategies. In the described scenario it is crucial to provide a solution that enables effective resource discovery in a highly heterogeneous and dynamic environment by allowing stakeholders to locate the available resources based on their specific requirements. On this basis, HADR management operators should be able to locate and interact with the existing and available IoT assets in a quick manner, thus calling for innovative solutions capable of addressing the large heterogeneity that usually characterize IoT resources in Smart City environments.

More specifically, the location- and context-aware nature of Smart City services provides the opportunity to implement the discovery of IoT assets within an *IoT domain*, i.e., a location (either geographically or administratively defined) in which it resides. In fact, most IoT applications following the edge computing paradigm are based on service components that execute in proximity to IoT information sources. Those applications will be best served by solutions that limit the scope of the discovery process to the resources in the current IoT domain.

Therefore, especially for HADR operations, it is necessary to consider a broader scope of application that will be likely to leverage IoT resources and service components in more than one IoT domain. For instance, depending on the headquarters's location, military or local enforcement would need to gather information for different locations within the city's existing infrastructure. As a result, it is necessary to implement solutions that enforce strict and application/resource type/domain specific information management policies for multi-domain IoT asset discovery - while at the same time implementing automated replication of information to improve performance. Finally, the significant heterogeneity of IoT assets and the broad scope of IoT applications and services call for the adoption of resource discovery solutions that are capable of supporting different communication and discovery protocols.

III. MARGOT

MARGOT aims to simplify the discovery of IoT devices and the development of IoT applications by providing a domain- and context- aware resource discovery service though a standardized M2M compliant interface. Fig. 1 depicts the architectural design of MARGOT by highlighting its major modules: Discovery Agents (DAs), the Information Processor, Federation Services, and the ReST API, as well as their interactions.

DAs are independent software modules that proactively or reactively discover and register the assets available within the IoT domain. To create a more adaptive and extensible architecture, we designed DAs as pluggable components, thus allowing the integration of new discovery protocols in a modular and extensible fashion. More specifically, as shown in Fig.1, each agent implements a communication protocol, such as MQTT or CoAP, and exploits its corresponding discovery procedure or protocol mechanisms to locate and interact with the available IoT resources. Furthermore, DAs store all the collected data on the internal MARGOT database, thus making the collected data available to MARGOT stakeholders. Behind the scenes, the actual discovery process is performed according to the protocol specific procedure. For example, the CoAP protocol provides a resource discovery mechanism that can leverage unicast when the IP of a particular resource is known or retrievable via DNS, or on top of multicast using a specific look-up message as described in [18]. Another example of a discovery procedure leveraging a different protocol mechanism is the use of wildcards in MQTT in order to instantaneously subscribe to all active topics and obtain information from all publishers in the network.

The discovery process can be executed proactively via discovery agents or after a specific trigger or event, such as a stakeholder request or a federated MARGOT instance request, in which case, the reactive discovery agents are invoked. The proactive discovery agents perform these discovery procedures periodically to address the continuous variation of available resources in the IoT domain. In fact, Smart City environments are characterized by the deployment of new assets, the downtime of constrained nodes, resource deactivation, and mobile device migration in IoT domains. Furthermore, in order to provide more effective proactive strategies, the rate of the periodic discovery process can be tuned based on the evaluated network churn (variation) ratio, thus avoiding a flood of discovery messages and unnecessary bandwidth consumption.

DAs are not limited to exploiting communication protocols to directly discover resources but can also behave as clients for already available cloud services installed to simplify the interaction with IoT resources. For example, regions like North America provide a service entitled 511, which exposes a M2M API that enables collecting data from traffic cameras, road events, and so on. In this way, MARGOT can cooperate with existing services in order to push data toward the edge network or even enhance the remote services API though internal elaboration of the data acquired.

In fact, the MARGOT Information Processor is responsible for elaborating the collected data and generating a comprehensive view of the network. In addition, the Information Processor acts also as a controller to tune the behaviour of the other MARGOT modules. Therefore, the Information Processor continuously analyzes the discovery results to recognize possible asset taxonomies and evaluates the network churn to increase the overall effectiveness of the components. Taxonomies are useful information that can be retrieved by the stakeholders to have more a comprehensive view of the network while the network churn ratio is used as internal feedback to regulate the periodic discovery processes. Finally, the MARGOT Information Processor is also responsible to coordinate the database replication operations.

Information processing allows possible MARGOT stakeholders to fetch more elaborated data and perform more sophisticated queries that can satisfy their specific interests. For example, consider a face recognition application designed to find a person during a HADR operation. In this case, the face recognition application can interrogate MARGOT and specify its interest in collecting information about available cameras within the city limits. To request such information, MARGOT provides a rich ReST API to standardize data access and also provides services to manually register available resources that might not implement any discovery protocol.

MARGOT also provides a specific internal module, called the *Federation Service* [17], that allows different MARGOT instances to federate. Through this federation process, MARGOT is able to exchange information and forward queries to other MARGOT instances, thus allowing stakeholders to discover resources available across multiple IoT domains. Furthermore, a federated MARGOT instance can adopt policies to automatically replicate important parts of the stored information to other federates. This capability will result in a distributed database where information of common interest is disseminated among the network to be quickly available, by means of the interest-based dissemination provided by MARGOT. Mechanisms are also available to limit the information being replicated, for example, based on geographical bounds. Finally, information exchange policies can also be enhanced to define different permission levels for each federate, i.e., information is shared only if the remote domain is entitled to receive that specific information [21].

IV. Experimental Results

We experimentally evaluated MARGOT within an emulated network using the Extensible Ad-hoc Networking Emulator (EMANE). The testbed is composed of 20 nodes connected by network links, which present a latency varying from 40 to 100 milliseconds. Each node plays a specific role in the experiment: sensor, client, or gateway. Sensors nodes represent IoT devices that generate data and communicate using a specific protocol, e.g. CoAP. Clients represent nodes interested in discovering all or part of the sensors in the network. Finally, gateways are dedicated nodes running centralized services, e.g. information brokers. Therefore, the emulated network counts a total of 14 sensors (7 CoAP and 7 MQTT), 5 clients, and 1 gateway node. Furthermore, we also present the effectiveness of MARGOT in terms of information prefetching and interface enhancement by connecting our emulated network to a Cloud service, namely 511ny.org[1] that provides information about public traffic camera in New York City, NY, USA.

CoAP sensors communicate via TCP, UDP and UDP Multicast to allow multicast discovery. These resources were initialized with the attributes defined in the CoRE link format that describes the sensor in terms of type, MTU, and the URI that can be used to retrieve the data. We defined these attributes in order to enable clients with specific requests to perform context aware resource discovery, if possible, and to enable MARGOT to identify possible resource taxonomies.

MQTT sensors connect and periodically publish data to the active broker running on the gateway node. Each sensor has a different transmission frequency that varies between 4 and 12 seconds. Sensors do not share the topic except for the part of the topic describing the resource type. For example, `/nodemqtt5/companyA/camera` or `/companyB/temperature/roof`. The topic naming scheme might be different from node to node in order to simulate the presence of different owners, who may adopt mismatching naming schemes.

Client nodes on the other hand were either applications implementing the mechanisms required to retrieve information about the available sensors in the network (client A, B, and C) or that interrogate MARGOT to fetch information about discovered sensors (client D and E). The gateway is instead a node responsible to run a MARGOT instance and the MQTT broker (in this experiment Eclipse Mosquitto[2]).

The clients that do not use MARGOT instead rely on the intrinsic mechanisms provided by the communication protocol to

[1] 511ny.org, available online at https://511ny.org/

[2] Eclipse Mosquitto, available online at https://mosquitto.org/

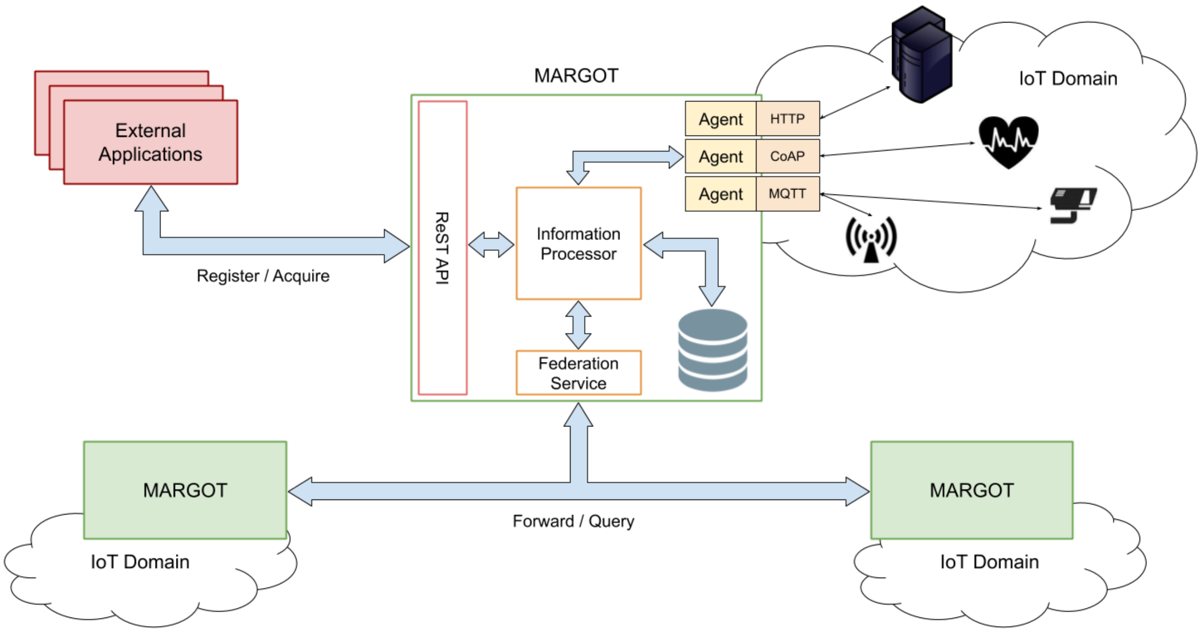

Fig. 1. The MARGOT Architecture.

retrieve information about the available sensors. More specifically, to discover the CoAP sensors, Clients A, B and C perform a multicast request to the URI `/.well-known/core`, which is the URI specified by the protocol to retrieve information about the registered resources on the CoAP server. We chose to adopt multicast, since the clients do not have prior knowledge about the addresses of each sensor. MQTT sensor discovery instead was performed by exploiting the MQTT wildcard mechanism. In fact, by subscribing to the topic `#`, each client will receive all messages that MQTT sensors publish. Then, we stopped this process when at least one message per topic has been received. Both CoAP and MQTT procedures are also performed by MARGOT DAs to discover the sensors of this experiment.

The results we obtained compare the total time required by each client to discover the sensors in the emulated network. To do that, we determined the cumulative numbers of sensors that were discovered over time in order to highlight when each client acquires information about each sensor. Fig. 2 depicts the necessary time for each client to obtain information about all the available CoAP resources. Clients leveraging on the CoAP discovery mechanism require between 400 to 700 milliseconds to have a complete list of the available CoAP sensors. On the other hand, clients that requested the list of available sensors from MARGOT obtained the same results in approximately 200 milliseconds. In addition to MARGOT clients having a shorter wait time, what is more interesting is the capability to obtain an instantaneous snapshot of the available sensors in the network. For instance, in a real scenario, Clients A, B, and C would have to indefinitely wait for discovery replies given that the number of sensors is unknown a priori. Furthermore, we show the results from Client E that performed a filtered request to MARGOT and thus received information about only the resources that fit its specific interests (four in this experiment), without the need for filtering the response locally.

The time required by each client to discover every MQTT sensor is shown in Fig. 3. To collect these results, the clients have been started asynchronously in order to highlight different results that our scenario might present. In fact, due to the passive nature of the discovery mechanism the overall time to discover the MQTT sensors depends on the sensor publication frequency and when the time when a client subscribes to all topics. The longest time required to receive information about all publishers is 9.1 seconds but in the worst case it might take until 12 seconds to discover every single MQTT resource since it is the longest publication interval among all the publishers. Clients that instead use MARGOT to discover the MQTT sensors waited only about 200 milliseconds. As in the CoAP case, clients that adopted MARGOT obtained complete information about the MQTT resources without any uncertainty based on the time when they subscribe. Furthermore, when using MARGOT, the clients do not need to subscribe to topics that do not fulfill their interests. This is even more evident for client E that performed a request limited to all the topics containing a specific keyword.

Finally, in Fig.4 we present the time that nodes required to interrogate a remote cloud service to obtain information about traffic cameras. In this case, we used 3 clients, of which only one is directly connected to the remote service. Fig. 4 shows the HTTP request performed by Client A has been resolved in 6.9 seconds by the remote service. This period is not just related to the network latency but also to the computation time required by the remote service to fetch all the requested information (there were 1290 cameras available). Instead, the clients connected to MARGOT received a response within 200 milliseconds. As shown for the other experiments, Client E performed a filtered request to MARGOT and thus was served with limited amount of information. The same result would be impossible in other ways since the remote service does not provide methods to perform specific requests, e.g. cameras available in a specific area, again underlying MARGOT's effectiveness.

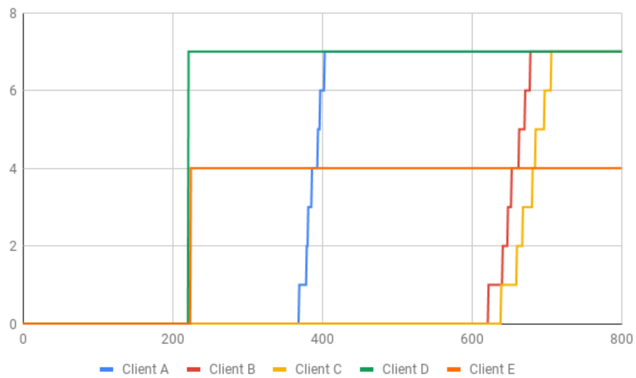

Fig. 2. CoAP discovery time (in milliseconds).

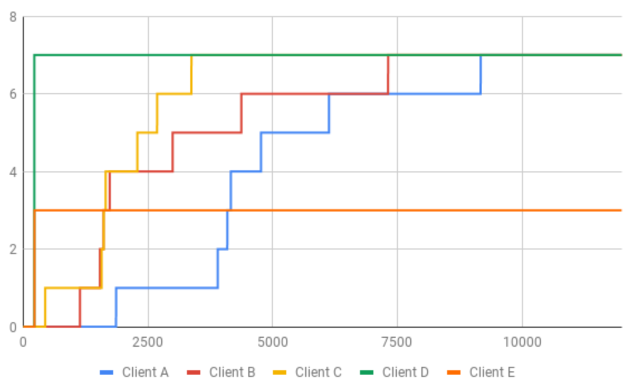

Fig. 3. MQTT discovery time (in milliseconds).

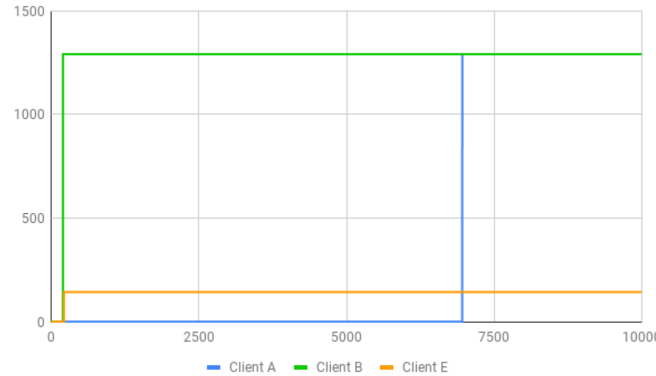

Fig. 4. HTTP discovery (in milliseconds).

V. Related Work

Resource discovery for IoT applications is still an open research topic. An interesting survey of discovery technologies for the IoT is presented in [23], where different solutions such as multicast DNS (mDNS), multicast CoAP, the Simple Service Discovery Protocol (SSDP), and others are presented and evaluated. In [24], the authors present a Named Data Networking (NDN) based solution for the edge network. This work proposes a discovery mechanism based on a service-response model. In this model, a consumer asks for a desired service to the devices in the neighborhood using a broadcast message with a pre-defined TTL. If a service provider is not found, the consumer sends another message with an increased TTL until a provider for the service is found or the maximum desired TTL is reached. On the other end, a provider will reply back to the consumer if it is eligible to satisfy its request. Finally, the authors propose a deferral scheme to avoid collisions over the same service request. Instead, in [25], the authors describe a discovery approach based on an interoperability model to bridge an NDN and an IP network. This approach makes use of mDNS inside the IP network and the Named Publish Subscribe Networking protocol on the NDN network to discover consumers and devices. To bridge the different communication protocols the authors propose the adoption of a gateway solution called Future Internet eXchange Point (FIXP). In [15], the authors describe a discovery approach that makes use of MQTT to keep track of publishers/advertisers (IoT devices) in a IoT-Fog environment. In particular, the authors propose a protocol, namely Smart and Power Efficient Node Discovery Protocol (SPEND) as a solution to create a reliable and energy efficient discovery solution for IoT applications. The protocol is evaluated via experimental results to ensure its power efficiency and effectiveness, and the authors conclude that MQTT is a reliable and efficient protocol for constrained devices.

Another interesting work is [26], where the authors discuss the application of IoT service discovery to ICN by proposing a semantic matching mechanism to achieve a more flexible discovery process. The described solution considers four entities: clients that are interested in receiving particular information, service providers that provide one or multiple services, a discovery Broker that stores the information about services and receives queries from users, and a semantic search engine, which performs the matching of query and services. In [16], the authors propose a distributed discovery mechanism for Internet of Things. This proposed solution makes use of the CoRE Resource Directory (RD) and CoAP as a standard interface for the discovery of and access to resources. In the proposed solution, IoT Gateways are responsible for implementing the RD within the single IoT domain. Finally, to enable global discovery across multiple IoT domains, the author describe a Distributed Hash Table (DHT) approach to build a global and distributed RD.

MARGOT differs from these other solutions by proposing a distributed architecture of gateways capable of re-routing queries and information over an extended network using query forwarding policies and to provide data replication policies and permissions. Furthermore, unlike other solutions, MARGOT defines the concept of pluggable discovery agents to enable the discovery and management of resources using a wide range of discovery protocols, thus not limiting its capabilities to a single discovery standard or protocol.

VI. Conclusion

This paper described MARGOT, a dynamic and distributed resource discovery and management solution for HADR operations in Smart City environments. MARGOT enables distributed queries for devices and resources by means of a federated architecture of MARGOT instances. Furthermore, MARGOT adopts distributed caching policies to speed up the resource discovery process. Finally, the modular architecture of MARGOT enables extension of its capabilities by plugging multiple discovery agents in order to support multiple discovery and communication protocols.

References

[1] Bandyopadhyay, Debasis, and Jaydip Sen. "Internet of things: Applications and challenges in technology and standardization." Wireless Personal Communications 58.1 (2011): 49-69.

[2] Saha, Himadri Nath, Abhilasha Mandal, and Abhirup Sinha. "Recent trends in the Internet of Things." Computing and Communication Workshop and Conference (CCWC), 2017 IEEE 7th Annual. IEEE, 2017.

[3] Lin, Jie, et al. "A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications." IEEE Internet of Things Journal 4.5 (2017): 1125-1142.

[4] Hui, Terence KL, R. Simon Sherratt, and Daniel Díaz Sánchez. "Major requirements for building Smart Homes in Smart Cities based on Internet of Things technologies." Future Generation Computer Systems 76 (2017): 358-369.

[5] Wollschlaeger, Martin, Thilo Sauter, and Juergen Jasperneite. "The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0." IEEE Industrial Electronics Magazine 11.1 (2017): 17-27.

[6] Atzori, Luigi, Antonio Iera, and Giacomo Morabito. "The internet of things: A survey." Computer networks 54.15 (2010): 2787-2805.

[7] Santana, Eduardo Felipe Zambom, et al. "Software platforms for smart cities: Concepts, requirements, challenges, and a unified reference architecture." ACM Computing Surveys (CSUR) 50.6 (2017): 78.

[8] Yaqoob, Ibrar, et al. "Enabling communication technologies for smart cities." IEEE Communications Magazine 55.1 (2017): 112-120.

[9] Petrolo, Riccardo, Valeria Loscri, and Nathalie Mitton. "Towards a smart city based on cloud of things, a survey on the smart city vision and paradigms." Transactions on Emerging Telecommunications Technologies 28.1 (2017): e2931.

[10] Shi, Weisong, and Schahram Dustdar. "The promise of edge computing." Computer 49.5 (2016): 78-81.

[11] Riberto, G., Govoni, M., Stefanelli, C., Suri, N., Tortonesi, M., "Leveraging civilian IoT infrastructures to support warfighting activities in urban environments", pp. 118-123, doi: 10.1109/WF-IoT.2018.8355184.

[12] N. Suri et al., "Analyzing the applicability of Internet of Things to the battlefield environment," 2016 International Conference on Military Communications and Information Systems (ICMCIS), Brussels, 2016, pp. 1-8, doi: 10.1109/ICMCIS.2016.7496574.

[13] Kott, A., Swami, A., West, B., "The Internet of Battle Things", IEEE Computer, vol. 49, pp. 70-75, 2016, doi: 10.1109/MC.2016.355.

[14] S. Russell and T. Abdelzaher, "The Internet of Battlefield Things: The Next Generation of Command, Control, Communications and Intelligence (C3I) Decision-Making," MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM), Los Angeles, CA, 2018, pp. 737-742, doi: 10.1109/MILCOM.2018.8599853.

[15] Venanzi, R., Kantarci, B., Foschini, L., & Bellavista, P. (2018). "MQTT-driven sustainable node discovery for internet of things-fog environments." IEEE International Conference on Communications, 2018-May.

[16] G. Tanganelli, C. Vallati and E. Mingozzi, "Edge-Centric Distributed Discovery and Access in the Internet of Things," in IEEE Internet of Things Journal, vol. 5, no. 1, pp. 425-438, Feb. 2018. doi: 10.1109/JIOT.2017.2767381.

[17] Lenzi, R., Benincasa, G., Casini, E., Suri, N., Morelli, A., Watson, S., Nevitt, J., (2013). "Interconnecting tactical service-oriented infrastructures with federation services", Military Communication Conference (MILCOM), 2013.

[18] Z. Shelby, K. Hartke, and C. Bormann, "The constrained application protocol (CoAP)", Internet Engineering Task Force, Fremont, CA, USA, RFC7252, Jun. 2014.

[19] Pradhan, Manas, et al. "Toward an Architecture and Data Model to Enable Interoperability between Federated Mission Networks and IoT-Enabled Smart City Environments." IEEE Communications Magazine, 16 Oct. 2018, pp. 163–169.

[20] Fielding, Roy Thomas (2000). "Architectural Styles and the Design of Network-based Software Architectures", Ph.D.). University of California, Irvine, USA.

[21] Poltronieri, F., Campioni, F., Lenzi, R., Morelli, A., Suri, N., Tortonesi, M. (2018). "Secure Multi-domain Information Sharing in Tactical Networks", 2018 IEEE Military Communications Conference (MILCOM 2018), NATO Special Track, 29-31 October 2018

[22] M. Tortonesi, M. Govoni, A. Morelli, G. Riberto, C. Stefanelli, N. Suri, Taming the IoT data deluge: An innovative information-centric service model for fog computing applications, Future Generation Computer Systems, 2018, https://doi.org/10.1016/j.future.2018.06.009.

[23] Bröring, Arne & Datta, Soumya Kanti & Bonnet, Christian. (2016). 6th "A Categorization of Discovery Technologies for the Internet of Things." International Conference on the Internet of Things: 131-139.

[24] Amadeo, M., Campolo, C., & Molinaro, A. (2016). "NDNe: Enhancing named data networking to support cloudification at the edge." IEEE Communications Letters, 20(11): 2264-2267.

[25] Quevedo, José & Ferreira, Rui & Guimarães, Carlos & Aguiar, Rui & Corujo, Daniel. (2017). "Internet of Things Discovery in Interoperable Information Centric and IP Networks: IoT Discovery in Interoperable Information Centric and IP Networks." Internet Technology Letters.

[26] José Quevedo, Mário Antunes, Daniel Corujo, Diogo Gomes, Rui L. Aguiar, "On the application of contextual IoT service discovery in Information Centric Networks", Computer Communications, Volumes 89–90, 2016, Pages 117-127, ISSN 0140-3664, https://doi.org/10.1016/j.comcom.2016.03.011.