López Escobar *et al.*
Journal of Cloud Computing (2022) 11:81
https://doi.org/10.1186/s13677-022-00354-x

Journal of Cloud Computing: Advances, Systems and Applications

RESEARCH

Open Access

In-depth analysis and open challenges of Mist Computing

Juan José López Escobar[1*], Rebeca P. Díaz Redondo[2] and Felipe Gil-Castiñeira[1]

Abstract

The advent and consolidation of the Massive Internet of Things (MIoT) comes with a need for new architectures to process the massive amount of generated information. A new approach, Mist Computing, entails a series of changes compared to previous computing paradigms, such as Cloud and Fog Computing, with regard to extremely low latency, local smart processing, high mobility, and massive deployment of heterogeneous devices. Hence, context awareness use cases will be enabled, which will vigorously promote the implementation of advantageous Internet of Things applications. Mist Computing is expected to reach existing fields, such as Industry 4.0, future 6G networks and Big Data problems, and it may be the answer for advanced applications where interaction with the environment is essential and lots of data are managed. Despite the low degree of maturity, it shows plenty of potential for IoT together with Cloud, Fog, and Edge Computing, but it is required to reach a general agreement about its foundations, scope, and fields of action according to the existing early works. In this paper, (i) an extensive review of proposals focused on Mist Computing is done to determine the application fields and network elements that must be developed for certain objectives, besides, (ii) a comparative assessment between Cloud, Fog, Edge, and Mist is completed and (iii) several research challenges are listed for future work. In addition, Mist Computing is the last piece to benefit from the resources of complete network infrastructures in the Fluid Computing paradigm.

Keywords: Mist Computing, Fog Computing, Edge Computing, Fluid Computing, Internet of Things (IoT), Integrative Literature Review

Introduction

The mass deployment of Cloud services has been a significant achievement in the last decade to enable flexible and transparent computing for almost any application. However, its centralized nature is not suitable for many applications requiring a real-time reaction, such as industrial control systems, smart intersections, etc. That is the reason new computing paradigms [1], such as Fog and Edge Computing, have been developed to bring services closer to the end-user, while reusing the existing resources in the network infrastructure. Recently, a new post-Cloud paradigm called Mist Computing has been coined to move computing power to IoT devices.

Mist Computing has emerged, as usual in the Internet technology trends, quietly, and it still needs much more work and time to become a mature and clear technology that revolutionizes the industrial ecosystem. Despite this, there exist many fresh research lines which leverage this paradigm, either focusing on a concrete topic or proposing a complete solution, as well as some Fog/Edge-oriented projects which touch on this topic unintentionally. A detailed review is required to understand the different proposals, detect synergies among them, and leverage knowledge that may set the basis for a future computing boom.

More specifically, the idea of Mist Computing arises from the return of Wireless Sensor Networks (WSNs) applied to the IoT, aspiring to give them specific

*Correspondence: juanjo@gti.uvigo.es

[1] Information Technologies Group (GTI), atlanTTic Research Center, University of Vigo, 36210 Vigo, Spain

Full list of author information is available at the end of the article

characteristics of Cloud Computing. Even though there have been proposals that underline the importance of this matter in data analysis for healthcare systems [2], advanced robotics [3], or smart traffic coordination [4], its formal definition is relatively vague and needs precise explanation to agree about it. However, Mist claims a change in the way IoT devices have been regarded in Cloud and Fog architectures as simple information collectors. With the Mist computing paradigm, IoT devices also take advantage of resource exploitation, location awareness, and smart cooperation among parties in order to create more scalable, autonomous and secure systems. Moreover, it signifies a decisive impulse in the IoT field, since it will address existing issues such as lightweight resource virtualization in constrained devices [5], dynamic service composition [6], distributed data fusion algorithms [7], etc. In essence, the Mist leads to improvements in technical performance aspects that will enable the progression of real-time and autonomous IoT use cases. It allows reducing the latency to make faster responses, supports high throughput demand from plenty of heterogeneous devices by not requiring sending the information to a centralized cloud location, and at the same time data is handled in compliance with strict privacy patterns.

Thus, this study seeks to highlight the importance of Mist Computing in the upcoming years and to give an initial comprehensive insight for future work lines. In addition, it tries to clarify the difference between Mist and other post-Cloud paradigms, since there is haziness regarding these terms [8]. In order to do so, a neutral discussion about Mist Computing to profile a subject-related taxonomy is addressed, which will help to distinguish their application areas and to identify new avenues for research, while serving as a referral point of Mist Computing principles.

The developments presented in this paper are aligned with the objectives of pushing intelligence to the edge nodes, upgrading ubiquitous computing to make constraint devices cooperate, and orchestrating on-demand services from the bottom to the top of the network. All that will be crucial in future 6G networks [9] and a breakthrough in the Industrial Internet of Things (IIoT) [10] to achieve extremely low latency and preserve data privacy. Consequently, this paper wants not only to synthesize previous works about Mist Computing but also to establish a common point to design a framework that sets the basis for the empowerment of use cases in the aforementioned scenarios and to identify challenges that need to be overcome. Namely, to identify the aspects that have not yet been addressed and the points of improvement, such as the potential synergies with complementary technologies, or the promotion of a versatile and accurate usage of each cloudy computing paradigm according to the needs of each scenario.

In summary, this research survey intends to clarify the Mist Computing concept and its principles for the scientific community and the suitability of each Cloud and post-Cloud computing paradigm for each use case. Furthermore, it identifies the open research topics related to the Mist scope.

The paper starts with a brief overview of the new paradigms that place the computing resources closer to the end-user and a strong justification of the need to perform this study. Then, in Section 2 an introductory explanation about Mist Computing, its roots, and its implications is done. The Section 3 describes how the review work has been performed, and Section 4 recognizes the topics which have been addressed by grouping together the relevant sources by subject. Then, Section 5 reviews the aspects that characterize the most popular computing paradigms according to the information presented in the previous part. In Section 6, an exhaustive analysis of the objectives and future research lines to be pursued to enhance Mist Computing is performed and finally, in Section 7, some final thoughts about the work are added.

The origins of Mist Computing

After the global adoption of Cloud Computing, researchers and enterprises have been working on additional strategies that place computing not so far from the final users to turn groundbreaking applications into reality, to give a better QoS, to exert greater control on private data, and to reuse resources as much as possible. They are usually grouped in post-Cloud paradigms [1], whose beginnings can be traced back to 2012 when Cisco started to talk about Fog Computing [11]. Figure 1 illustrates the evolution and roots of most common computing paradigms, including Mist Computing. Thus, we can roughly define Mist Computing as pushing cloudy computing properties downward to sensor networking at the extreme edge of the network. In particular, Fig. 1 shows, using color-coding for each computing paradigm, when each term was coined and when they started being adopted in real scenarios. In addition, the Figure displays the evolutionary relationships between the different paradigms. Nevertheless, even though several years have passed, there is still no clear consensus among authors about the limits and implications with regard to Fog, Edge, Cloudlets, and Mist. Thus, this publication aims to provide an insight into the nature of the different paradigms.

The Fog appears as an evolution of the Cloud to bridge the gap between the computing resources and end-users in order to reduce response time, progress on compatibility between heterogeneous entities, and increase

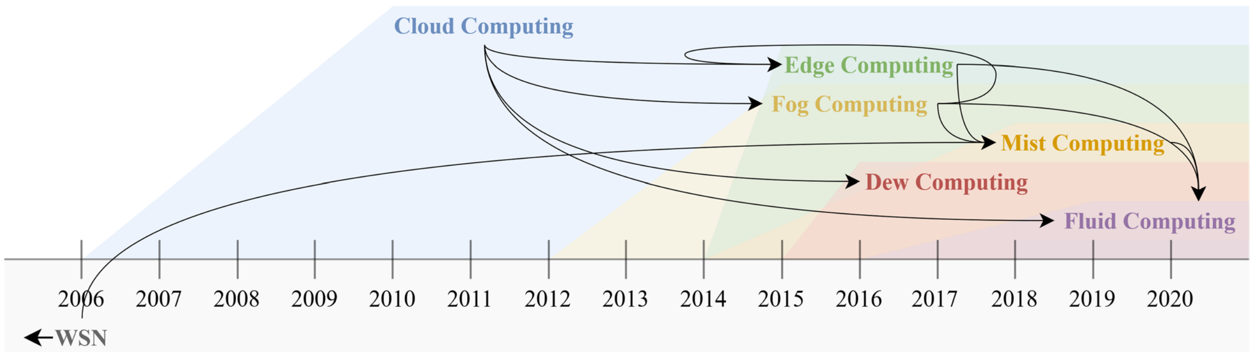

Fig. 1 Evolution of Cloud and post-Cloud paradigms

scalability [12]. It was initially conceived as a set of technologies to enrich the IoT world and work independently of the Cloud, but now it may apply to any company infrastructure that extends, via virtualization, the computation, storage, and networking services of the Cloud to a place near the access network. It has been standardized by the OpenFog Consortium to promote industry automation, collaboration, and interoperability.

The term Edge Computing may be slightly confusing because it was originally associated with CDNs (Content Delivery Networks) and then, telecom operators adopted the name to refer to processing done at the edge of the operator network with the MEC (Mobile or Multi-access Edge Computing) [13]. Furthermore, the same term is sometimes used for IoT gateways with processing capabilities [14]. Now, we can define Edge Computing as the processing that takes place in devices between the Cloud and the end-user device, including all the networking devices that handle the information that flows from the data source to the border of the Cloud. Then, depending on how the network is split among operators and organizations, the Fog may be included as part of the Edge, or vice versa (the Edge considered as a component located at the boundary of the Fog [1]), or even considered independent paradigms.

Additionally, any small-scale data center or cluster of computers on the Edge of the network providing Cloud services is known as Cloudlet, that is, virtual machines that offload processing of end devices and usually support mobility management [15]. They are the virtual components of the Fog that offer mobility support for resource-intensive applications [16], that is, they can be regarded as the MECs in a generic cloudy system.

Going back to the main focus of this work, the Mist might be understood as a lightweight version of the Fog that is located at the extreme edge of the network, as it is mentioned by the NIST [16]. The term was introduced in 2014 by Cisco, and it concerns collaborative pervasive computing at the extreme or outer edge of the network, that is, bringing computing and storage tasks to the sensors and actuators that are placed at the bottom layer of an IoT network.

Concretely, devices at the bottom layer are mainly low-power microcomputers and microcontrollers equipped with sensors and radio modules, and powered by batteries or alternative power sources. They may act as thin servers to exploit their compute, storage, and network resources to run varied code in a pseudo-virtual environment to suit the requirements of the applications.

More specifically, Mist nodes operate at the outer edge of the infrastructure, similarly to Cloudlets but on a larger scale, and are interconnected together. The Fog and the Cloud sides transparently. Thus, Mist takes IoT to the next level (compared to traditional Customer Internet of Things (CIoT), which assigns all data processing to the Cloud), shaping an autonomous system that will provide intelligence to IIoT. In addition, Mist architectures must be prepared to work independently of the Cloud, either because of potential interruptions in the connectivity or the need to use only the outer edge components for satisfying the latency requirements of the application, to avoid transmitting large amounts of information, etc.

Given this new scenario, diverse consequences arise related to performance metrics. In particular, latency is clearly impacted since the messages are processed very close to the origin (locally or distributed across the local network), preventing large delays caused by the Cloud. Indeed, using the Mist paradigm is a requirement for new use cases that would be impossible to implement in any other way in terms of throughput, consumption, and self-awareness. It also addresses the growing concerns of the society about the privacy of their data, which with Mist is processed and stored in controlled scenarios away from third-party Cloud infrastructures.

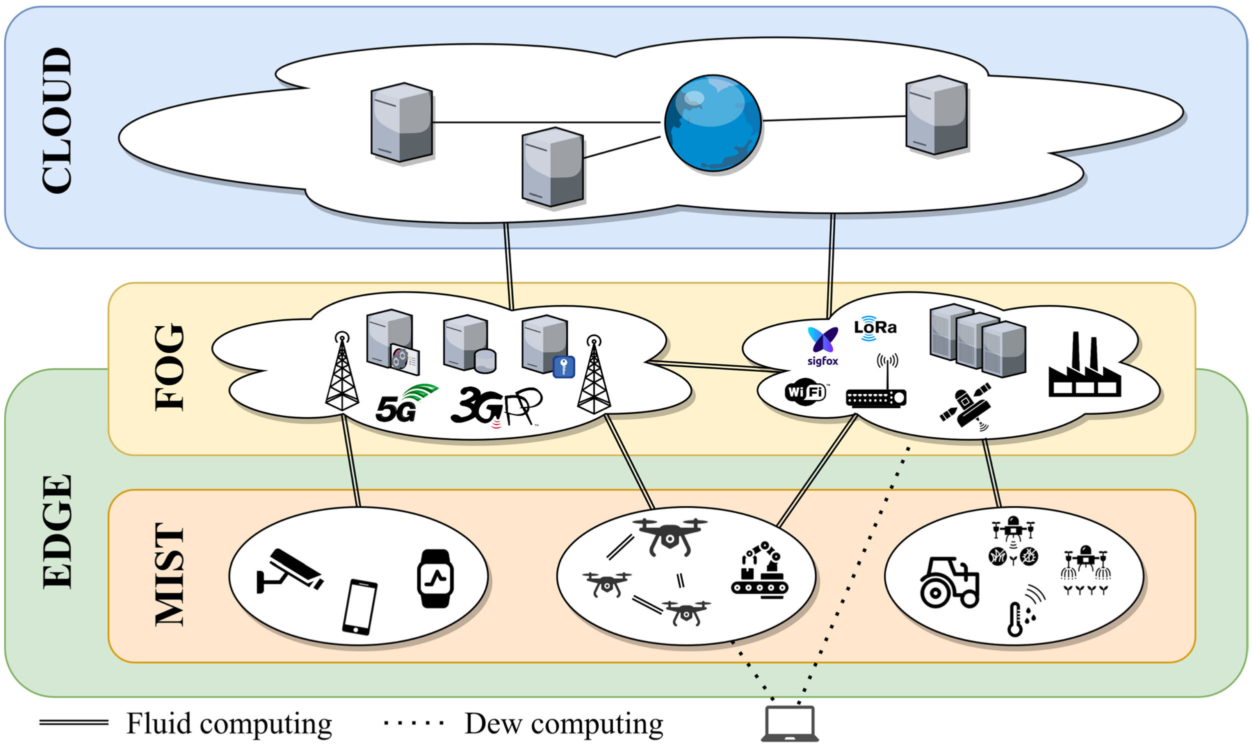

Fig. 2 Cloud and post-Cloud computing architecture

A similar contemporaneous concept is Dew Computing [17, 18], which might be confused with Mist, but it relates to offering Internet services to end-users without continuous Internet connectivity. That means having tiny instances of Internet nodes at the local network of final users, which provide a transparent experience for accessing Internet services. We can see both Mist and Dew paradigms as ubiquitous computing, but the latter one is strongly oriented to offering service to the human being.

As discussed, nowadays we have many choices to distribute the information processing tasks, having the possibility to take advantage of all the computing power in all the elements available in the network. All things considered, we can adopt the idea of Fluid Computing [19, 20] to link together Cloud and post-Cloud computing paradigms. It refers to unifying all through virtualization of computing, storage and networking resources regardless of the location of the application, in order to attain a cooperative end-to-end architecture as shown in Fig. 2. Its main objective is to offload tasks to the proper computing layer lively regarding the real-time requirements and the status of the network infrastructure to offer the best possible service and to optimize use of available resources.

Figure 2 shows how each computing paradigm, using the same color code as in Fig. 1, corresponds to elements in a straightforward separated location. The Cloud corresponds to a high-capacity grid of computers that offer transparent operation. Moving down to the Edge, computing performance is being limited, but the quality of service is improved, since we move to the Edge, devices are constrained in terms of resources and power. Thus, algorithms must be adjusted to fit the new environment and use the resources efficiently, otherwise running services will benefit from shorter connection paths that attain lower response time, handling flows of data from vast number of devices, and data can be confidently managed in the local context for the surrounding devices. Additionally, the Mist differs from Fog in the interaction with elements in the direct network and the fact that data can be retained in the local environment. It is the result of applying features from cloudy computing paradigms to WSN technology. This differentiation is further developed in the following parts of the paper and a more detailed understanding is attained.

Methodology

Integrative Literature Review [21] is a good approach for identifying and studying the relevant topics in Mist Computing, a new and promising subject in the following years, since it only considers qualitative aspects that help to understand it. This technique, more popular in social sciences and humanities, will lead to renewing and improving reviews in the engineering sector using objective criteria to develop the research work. Hereafter, the followed steps and criteria to select pertinent sources are exposed to allow proper validation of the work done.

The first point to take into account is to find the right terms for the search, thus we have chosen them to focus on Mist and Edge Computing, post-Cloud architectures, and network communication in IoT so that the most key sources can be retrieved. Besides, another concept that has been discussed is the elaboration of a middleware layer up to the Cloud/Edge/Fog/Mist architecture to enable fluid computing, as has been mentioned previously. Therefore, the specific combinations of words and their justification are:

A. *mist AND (fog OR edge) AND cloud AND computing.* It tries to retrieve works that explicitly regard Mist computing as the extreme edge of Fog or Edge Computing. The word 'cloud' has been included to focus on the desired topic.
B. *mist AND cloud AND computing AND (middleware OR architecture).* It fetches research proposals of network middlewares or architectures directly related to Mist Computing.
C. *post-Cloud AND iot.* Lastly, a search about post-Cloud paradigms oriented to IoT has been included because they are essential in these new technologies.

In order to optimize search effort and reduce the potential number of results, we have decided to filter by date from 2015, since the previous year Cisco started to talk about Mist Computing and the bulk of firm proposals arrived a little later in time. Nevertheless, this does not preclude the punctual inclusion of previous works that are somewhat previous or posterior but are clearly helpful.

Otherwise, database lookup has included the following scientific and technological sites that index journals and books related to electrical engineering and computer science: IEEE Xplore, Springer in Computer Science and Engineering disciplines, ScienceDirect filtering by Computer Science and Engineering areas, ACM Digital Library, MDPI and Wiley in Computer Science and Electrical & Electronics Engineering. Furthermore, Google Scholar was used at the end since it gives a global view of the entire spectrum and may provide forgotten articles with the previous ones.

Moreover, other essential sources of information have been the references that appear in preceding reviews and assessed papers about related issues, as well as the new publications that cite the ones already included in the bag. This way, we have a deep understanding of all-time proposals.

The procedure to extract the most important proposals from all the search queries has been conducted in two phases. In the first stage, we have performed the corresponding queries in the databases, according to the necessary filters to tune them correctly and include only those papers that are better ranked by relevance score in the search engines and address subjects related to computing techniques at the extreme edge of the network in constrained devices. We have eliminated those elements that do not comply with the field after reviewing the content, so that the potential list is not excessively long. And secondly, a very comprehensive reading was carried out in the remaining ones to recognize which papers are truly useful for the research objectives of this work, and we have identified their original contributions and the scientific relationships between them. During this process, we also detected duplicated contents and explored the citations to refine the final references.

In view of all the explained procedure, we can state that the resulting scientific survey has a rigorous structure for identifying high-quality contributions and facilitating the dissemination of knowledge.

Analysis

In this section, the survey per se is depicted, following the fundamental principles of integrative literature review. First, the search and evaluation tasks are addressed to explain how the filtering process resulted in the final set of references. Then, an accurate review is fulfilled by following a critical sense, in the way that each individual topic is explained, and the different proposals are coherently interrelated.

In accordance with the search terms expressed in the previous Section, Table 1 outlines the number of references that have been considered from each bibliographic database. For each query, there are three columns that represent (*R*) the raw number of results retrieved, (*S*) the number of references that seemed to be relevant for the reviewed topic, and (*F*) the number of proposals that have been actually useful and finally included in the review, respectively. There is also a final row that shows the total number of papers that have been finally used from each database, taking duplicates into account.

Additionally, supplementary information targeting Mist Computing aspects, which were obtained from references and citations included in the previously selected papers, have been included for a full vision of the topic. In so doing, we are aware of documents that are relevant but that otherwise, they would have remained outside the scope if we followed only the static search criteria.

Table 1 Number of references retrieved during critical analysis for each search engine

Search Query	IEEE			Springer (CS)			Springer (Eng)			ScienceDirect (CS & Eng)			ACM DL			MDPI			Wiley (CS)			Wiley (Eng)		
	R	*S*	*F*	*R*	*S*	*F*	*R*	*S*	*F*	*R*	*S*	*F*	*R*	*S*	*F*	*R*	*S*	*F*	*R*	*S*	*F*	*R*	*S*	*F*
A	392	45	19	86	17	2	173	19	7	146	16	2	70	10	2	48	14	4	17	2	1	53	4	1
B	359	41	18	85	16	2	160	16	8	113	23	2	65	10	2	44	13	4	19	3	1	36	5	3
C	21	16	2	6	5	0	1	0	0	5	4	1	3	3	1	9	4	0	1	0	0	1	0	0
Total	23			2			9			4			3			4			1			1		

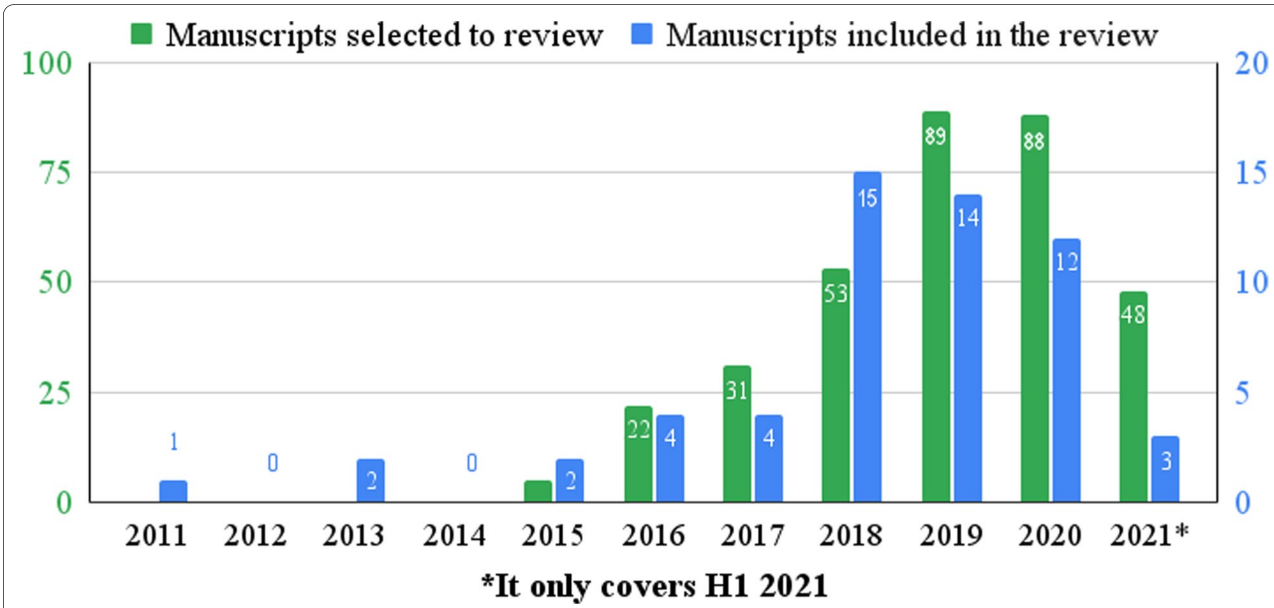

Fig. 3 Number of initially selected references (green) and actually included ones (blue) in the review per year

Finally, 47 papers have been picked out from the chosen scientific databases and 10 references have been identified using citation linkage, making a total of 57 papers. The final number of references may not seem many, but Mist Computing is at an early stage and this review technique allows retaining just highly pertinent sources.

As it is shown in Fig. 3, the number of selected manuscripts to evaluate in the review (green) is on an upward trend, and most of the reviewed articles (blue) have been published in the last few years. Thus, we can assert that the chosen time range was right, and the studied field of interest is rising. Besides, there are a few references prior to this date that were identified with a paper citation search and allowed us to find out that it had been vaguely envisaged at its origins. It also needs to be stressed that the review has been performed in the first half of the year 2021, and that is why references from this year are lower than expected.

The review is divided into the noteworthy topics that have appeared during a profound and careful reading of works related to Mist Computing architectures. We organized the topics in a top-down approach to help the reader to understand the concepts before going into more detail. First, generic topics regarding use cases and overall solutions are considered, and then particular solutions for each technical topic are addressed. Figure 4 outlines the topics to have an overall view of the work. It is important to note that the following references consist of subject-specific proposals, which have helped to group them correctly, and entire architecture solutions, which have picked the topics out.

Applications

A significant part to be discussed in any upcoming technology is the fields where it is going to be applied to and the situations that are going to be solved. In the case of Mist Computing, it will bring the possibility of making vanguard IoT settings feasible, and it will also progress on improving those intelligent environments. This section

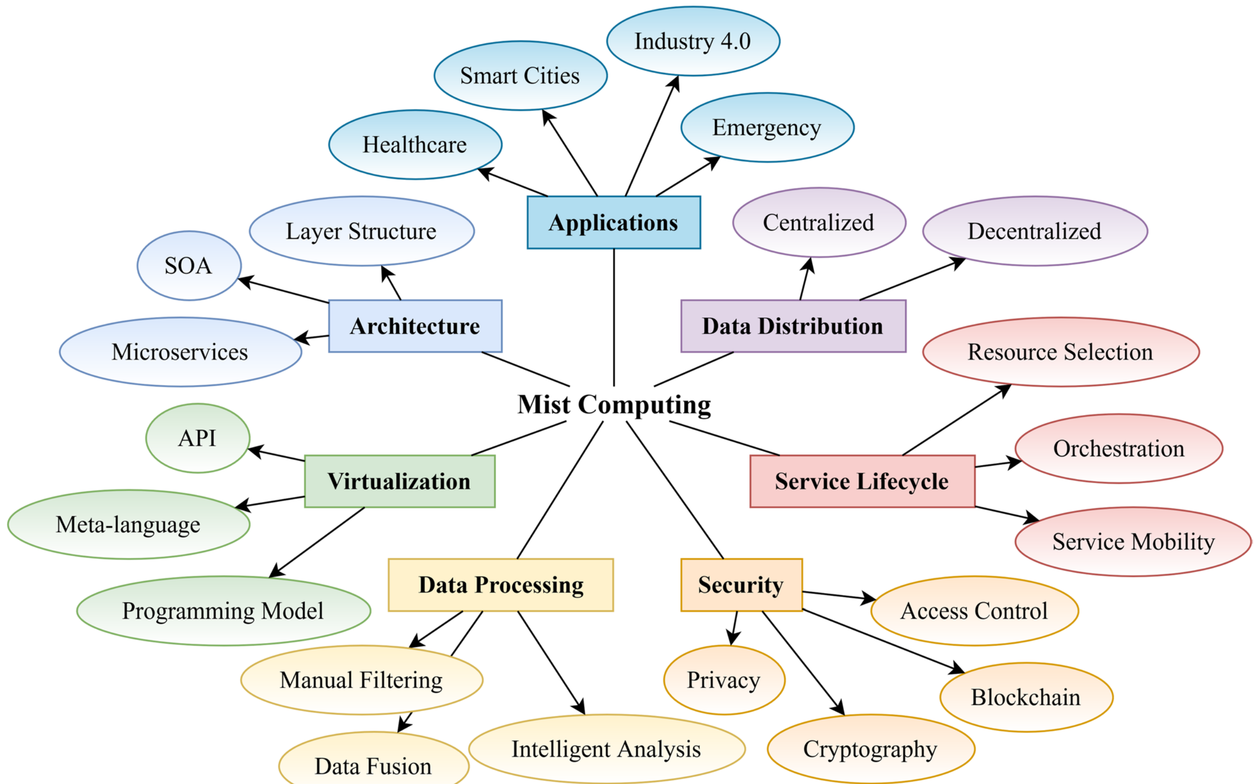

Fig. 4 Research working areas in Mist Computing

explores the possibilities provided by Mist Computing for the most relevant applications.

Due to the confidential treatment of data in healthcare systems [22], the Mist gives an opportunity to harness new IT frameworks [23] that overcome autonomously services to have complete patient monitoring [24]. This is known as the Internet of Health Things (IoHT). It also includes smart processing of data in order to maintain situational awareness. Detecting and signaling events must be done continuously since the environment changes [25]. In this connection, the large amounts of data from different medical centers provide an opportunity for Big Data systems where Federated Learning takes sides to share information between medical centers and improve the quality of their results [26–29]. That kind of system can be applied to many vital monitoring scenarios in order to improve the experience of the medical staff using augmented reality with big volumes of data [30] and enhance the quality of life in society, from hospitals [31] to elderly people homes [32].

Smart cities are envisaged to be an important focal point for the generation of large amounts of raw data to handle and draw conclusions. Extreme edge technologies will help to reduce data traffic within the network infrastructure [33]. To do so, sensors and actuators must be reinforced in their capacity to operate distributed computing tasks dynamically and cooperatively. This way they will be able to generate the desired data analytics at the edge of the network without centralized Cloud intervention [34]. Some examples of advanced smart city applications, which need to manage an enormous amount of devices and intelligently combine the data to get a prompt answer, are city surveillance and traffic monitoring, which are growing trends in big cities [35].

The revolutionary Industry 4.0 poses many logistic and manufacturing processes that increase productivity. Edge technologies must be embraced to ensure that they work as intended. Specifically, automation in production lines can be improved with machine vision capabilities in factory systems, such as robotic arms used for bin picking [3] or product quality check via multi-camera surveillance [36]. Such computer vision architectures can divide the processing between the Cloud and the components at the production plant to accelerate production. Another useful case study is indoor navigation to allow robots to move autonomously inside the building, identify people in an area or get geo-distributed analytics [37].

Furthermore, the Mist is regarded as cross-cutting technology that can be transferred to other fields like exploration and rescue in hazardous areas or even military settings. Of particular importance here are surveillance functions from a distributed infrastructure of cameras that collaborate to attain a faster and better-quality recognition of people and objects [38]. This is a very interesting case to use drones for patrolling activities, where they help one another to warn about an event.

In spite of the existence of running systems for various IoT application fields like vital monitoring, smart agriculture, smart cities, distributed data mining, or manufacturing control [39], the Mist Computing paradigm will enable fresh use cases in those scenarios and new ones. For instance, smart cities and factories will better manage the energy of smart lighting and production systems [40], and autonomous driving and military settings will be encouraged [41].

It is shown indeed that Mist Computing is applicable to many cutting-edge IoT scenarios like healthcare systems, smart cities, the fourth industrial revolution, and rescue and surveillance scenes thanks to the growing sensing setting, massive data generation, and data sharing between autonomous systems. It will also accelerate the adoption of new technologies such as artificial intelligence, mixed reality, and connected vehicles.

Architecture

Facing new computing paradigms implies new architectural models and modifications to existing ones to adapt to arising challenges. In the context of Mist Computing, one extra layer (sometimes divided in two for perception and computing capabilities) is considered at the bottom of the reference Cloud and Fog/Edge architecture. It usually offers its services in two different ways: extending the edge network and being dependent on the Fog and Cloud platform, or forming a detached layer that can operate independently.

Most of the architectures rely on an upper side infrastructure that helps with tasks that are out of the scope of the capabilities of mist devices, thus they need continuous vertical connectivity to accomplish a fully operational service. Such is the case of *EXEGESIS* middleware [35], which depends on a virtual Fog (vFog) infrastructure to interconnect Mist networks and Cloud services. Namely, from all regular mist nodes (RMN) in each Mist network, one node is appointed super mist node (SMN) to serve as a link to the vFog and to manage its neighborhood. This layered structure has already been contemplated in several final application scenarios, such as indoor surveillance systems [38] or smart healthcare frameworks [22, 23] which split the entire infrastructure into layers according to their functionalities. In order to normalize this full continuous computing strategy in the IoT world, the *IoTinuum* architecture [39] serves as a reference for developing different flavors of architectures. Here, we must emphasize the role of the IoT gateway, which not only bridges the Mist to the Fog or Cloud but also supports executing part of the Cloud/Fog business logic closer to the IoT network to satisfy real-time case requirements [36].

Conversely, another vision is to create a Cloud-less environment that operates autonomously and hosts all applications for end-users. As a result, services perform better quality because there is no need for a central core that could add delays, faults, and saturation to the system. A very promising project in this regard is the *Information Flow of Things* (*IFoT*) framework [42] which defines three layers to capture and process data, distribute information flows and render human and automated instruction recipes. This way, end-users in isolated areas can have comprehensive services running without Cloud connectivity [43].

In relation to the way how application services are composed and delivered to final users, the Service-oriented architecture (SOA) paradigm is an easy option to provide that access. For example, the *mePaaS* framework [44] is based on an Enterprise Service Bus (ESB) to execute a requester workflow in a plugin model using the components of a mist node. This methodology has also been used in the *SoA-Mist/MistGIS* framework [26, 27] to allow clients to connect to Mist services directly, and use Fog and Cloud platforms on on the back.

A lightweight and more flexible alternative to compose applications is the Microservice architecture, where independent and self-contained software modules are interconnected to build a distributed application. This concept is exploited in several research projects, for instance, the *Niflheim* middleware [45], where microservice units are designed to be deployed in all the tiers of the network and enable end-to-end collaboration between components. Also, the *nanoEdge* model [46] defines a nano-version for single-purpose functionalities, and in [47] microservices are implemented on top of CoAP technology.

All in all, novel IoT architectures are envisaged to have a high level of versatility in all nodes with respect to configuration and service operation. That is to say, the Mist layer must raise self-autonomy but it should also lean on upper layers to cope with the needs of all users, thus coordination among network parts must be ensured to adapt quality patterns to each situation empowering fluid computation. In addition, this degree of flexibility is reached by developing compact software units that can be deployed, combined, and moved easily depending on the state of the network, then new approaches inspired in microservices have to be investigated.

Data distribution

The way information is distributed along the entire network and understood by the *things* in the Mist layer is essential to attain flexible and efficient communication to ensure lightweight protocol design. Devices in the Mist need to have a common way to disseminate the gathered information and results achieved, and share with whoever needs them. Given the nature of these devices regarding network performance, we consider that the Publish-Subscribe pattern or a topic-defined data model would be more appropriate to disseminate knowledge, and it may be performed centralized or decentralized manner, as it is allowed in the ProWare middleware for self-aware health monitoring [25].

On one side, proto-Mist IoT scenarios usually rely on a central broker that stores published data, handles subscriptions to topics, and provides an abstraction to interoperability. This broker, generally based on Message Queuing Telemetry Transport (MQTT)[1], may be located in the Fog layer to reduce latency and increase scalability [38, 48] and even the Fogs from different Mist networks may be directly interconnected to create a single space, as it is done in the Application Agnostic Architecture framework [37].

On the other side, more advanced applications exploit the distributed architecture of new generation Internet networks based on Named Data Networking (NDN) fundamentals [49], which are seen as enablers of Fog and Edge smart use cases [50] since addressing and storing are combined to generate more flexible services. A similar locally limited approach is Data Distribution Services (DDS)[2], which establishes a virtual bus to deliver many messages with high performance and to control QoS, and it has been widely used in collaborative IIoT applications such as a smart platform for the energy industry [51]. Drawn from the DDS model, a novel technology called Zenoh[3] takes it further, and it defines a global Pub/Sub geo-distributed network for data in motion that supports data queries, reliable communication, and dynamic discovery. Moreover, following the key-value principle of NDN, in EdgeKV [52] a storage system based on Distributed Hash Table (DHT) for local networks interconnected by an upper one in the Edge is presented to provide a location-transparent infrastructure that facilitates final application development.

It can be seen that data distribution and location are key for highly efficient Mist communications, since smart *things* need to cooperate with other *things* in real-time, avoiding intermediate nodes. It is for those reasons that NDN-inspired architectures, where data is distinctly identified, and they immediately notice any updates, are more popular to adapt to data-centric networks, and distributed Pub/Sub architectures will be a boom in near IoT deployments since they can deliver data on-demand.

Virtualization

The interaction between heterogeneous devices entails a drawback in developing, deploying, and moving applications, which is more challenging in Mist computing because of the use of diverse microcomputers and microcontrollers with different architectures and limited resources. Consequently, approaches to define a sort of abstraction layer, which empowers unified access to resources and facilitates the developer's work, are seen with great interest in Mist environments and the upper application layers that may lean on.

A very common and powerful way to carry out such a task is the creation of an Abstract Programming Interface (API) which locates, above the hardware, a set of well-described programming calls that comprise all the functionalities. This requires a southbound and northbound operation to suit a specific platform and enable the use of those resources by external agents respectively, as it is suggested in *EXEGESIS* platform [35]. The API can be designed with many technologies, but the general idea is to enable transparent interaction with an infrastructure pool of IoT devices, as in *Niflheim* middleware [45], which provides a REST API to operate tiny *CerberOS* devices. Otherwise, the *Calvin* framework [53], and its constrained version [54], defines both sides of the interface as a Python API that is used by the actors that have been assigned to each hardware device. The same work has been pursued by the *FITOR* system [55]. Continuing with this approach, the *nanoEdge* model [46] describes a programming model to configure mobile agents that offer CoAP endpoints.

Another simpler method to deal with provision virtualization is the establishment of a standardized language that must be spoken by all sides. Some particular examples of this method in Mist environments are JSON-formatting for service management in the extended version of *mePaaS* framework [56], and to transmit data from sensors and instructions to actuators using the SenML specification [48]. Even choosing an adequate range of ontologies from the Semantic Web stack to describe the IoT hardware and their ecosystem [57] is a convenient solution. This approach results in a very light footprint impact in the abstraction layer.

In the light of all of this, a programming model must be established to operate easily over the infrastructure where resources are exposed and services composed. It

[1] Message Queuing Telemetry Transport https://mqtt.org/

[2] Data Distribution Services https://www.dds-foundation.org/

[3] Zenoh http://zenoh.io

must consider requirement constraints and ontological functionalities of tasks. A good way to do this is to define a layered structure that differentiates between hardware/resource and software/application sides, and that manages the provisioning, discovery, and configuration phases [58]. In this respect, several Fog models that are geared to services at the extreme edge have been developed both in the commercial and scientific sectors. *MobileFog* [59], and the extended version *Foglet* [60], is one of the first efforts to design a programming model that allows deploying an application decoupled from the IoT platform. It uses messages from APIs and event handlers to communicate the processes and compose the final service, making it possible to move the instances according to migration algorithms. Following the Dataflow programming paradigm, the *Distributed Dataflow* (DDF) framework [61] and the *FogFlow* model [62] allow the development of full services by creating and connecting lightweight flows that run in the exposed computing resources of the Fog/Edge according to a direct acyclic graph (DAG) design. A breakthrough proposal is introduced by *fog05* [63] since it uses a data-centric network to manage the resources of agents and to deploy entities that must have a state to orchestrate the final service.

So far, there exist several modes to exploit transparent access and interoperability in constrained edge devices, but they are characterized by a high level of complexity and a low level of flexibility. It is necessary to start working with them to create new architectures and platforms to provide new solutions for the simple distribution and deployment of applications. The need for a common framework that complies with numerous platforms is crucial, and it must give a unified entry point and meta-language to interact, as well as easy and transparent access to virtualized resources to allow adaptive usage following a predefined programming model.

Service lifecycle

At the time of operating a service in any distributed computing paradigm, several aspects must be addressed to offer transparency and flexibility in the access to resources and automatizing processes. This was already the case in the Cloud, and when moving down to the Edge new needs arise. In particular, the Mist poses extra challenges due to its essence regarding the mobility and limited performance of devices. For this reason, it is necessary to correctly manage the available resources to reach a good level of flexibility in the running services and get the most out of the existing network. Hereunder, three lines of action involved in this process are analyzed:

Resource selection

Since one aim of Mist Computing is to exploit resources in the entire network as much as possible, making a good choice of which elements will be used regarding service specifications, availability, location, and network status is a fundamental issue. Despite its significance, only a few relevant works have been explored to optimize this exercise. Theoretical models to rank node suitability for a kind of application and identify the best ones in a Cloud-Fog-Mist architecture (according to parameters like processing power, storage, energy consumption, neighborhood topology, etc.) have been developed in [64] and [65]. Another outstanding work targeting persistent storage is the *Fair Storage Distribution* (FSD) [66] algorithm, which composes a distributed file system for replication of sensor data using physical storage of extreme edge devices instead of the Cloud. This is done from a central node that controls the state of all edge devices (available memory and energy supply) to dynamically change the flows, according to the programmer's decisions.

Orchestration

Managing effectively the resources is crucial in advanced IoT environments, where a vast number of devices are into it, because it allows to know the network status, synchronize any topology change and come to a reactive decision to keep the desired application logic and quality. Besides, it can take action in composition and deployment phases to verify that applications use the pool of resources rightly [45] and it may have full knowledge of the resources in upper layers to use them if it were necessary [67], so the level of complexity may differ between designs. In order to facilitate the deployment of applications in any kind of device, it is recommended to break application services up into modular standalone fragments that can be easily allocated, interconnected, and administered according to needs at a particular time, like nano-services in *nanoEdge* platform [46]. In so doing, the deployment phase is streamlined regarding auto-configuration, dynamic scaling-up, and service discovery [47]. In terms of managing the running services, a simple approach is followed in *EXEGESIS* [35] where it designates a super node for each Mist network, which collects information about the resources and status of its neighborhood, to forward it to an upper layer entity that executes the adequate instructions. An alternative is to define two entities per Mist device, either on the orchestrator or in the devices themselves, in order to monitor available physical resources and running services, respectively [44, 68]. In the particular case of *mePaaS* framework, it has been upgraded to also be responsible for scheduling resources according to incoming requests to available services [56]. A more complex, but stronger, implementation is to customize Fog orchestrators to cover extreme edge devices like *FITOR* [55], which is inspired in ETSI MANO and can describe, deploy, monitor, and react to microservices according to hardware and QoS constraints. All those previous orchestration proposals have in common their centralized character, but in *Fog05* [63] a decentralized solution is presented to increase scalability and fault tolerance, and to take advantage of both Fog and Mist facets.

Service mobility

Considering the workload of the network or special issues in the services, the available infrastructure may vary, and it would be necessary to move some tasks to another place. This process is known as "offloading" and it can operate vertically, pushing processing from the Cloud to the Edge, and vice versa, or in the horizontal axis of the network layer, that is, between nearby devices of similar nature. The latter has grown in importance in Mist Computing because extreme edge devices can collaborate to compose powerful application services, maintaining low latency in response messages and reducing the network core usage [68]. A very easy way to conduct it is to fragment all services into tiny single-purpose functions that can be executed by any node in the Mist cluster, then, through a Pub/Sub system like MQTT, a device can request an operation which is answered by any of the potential collaborators [69]. This technique can also be used to deploy a service that has failed in a new running machine if this piece of software, known as Mobile Intelligent Agent [67], has saved the context, thus interruptions in the service experience can be avoided.

With the aim of managing deployed services in the existing infrastructure effectively, its lifecycle must be taken into consideration by the algorithms to choose the appropriate resources for the composition and coordination of services, through resilience adjustment, according to the required quality of service.

Data processing

One of the key points of Mist Computing is the ability of extreme edge nodes to execute sophisticated operations. This means that tasks, which would normally have been assigned to the Cloud or the Fog, are moved towards the bottom Edge. This enhancement of intelligence allows processing data in order to analyze, fuse and filter as intended, as well as to reduce the use of the core network, to guarantee reliable and low-latency communications and to ensure privacy regarding which information is shared. Thus, novel data analysis techniques have been adopted to extract precious information exploiting the decentralized and distributed nature of the Mist.

A usual task for the Mist is to identify valuable information from data generated by sensors in order to minimize network load and storage requirements and speed up future data analysis. This filtering typically depends on the specific sensor. Through techniques such as time-varying adaptive threshold [70] or manual policy rules [29], most of the noisy data is removed to acquire a good ratio of useful data. This second method has been applied in IoHT solutions where the perception layer generates raw sensitive data which needs to be pre-processed [22], and if it needs intense processing, data is forwarded to upper layers where powerful nodes can perform it [23].

In an attempt to leverage the capabilities of Mist devices with ongoing trends, some artificial intelligence technologies have been adapted to better analyze data, come to correct conclusions, and execute the right instruction in the actuators in meeting the latency needs. A clear example of this is the implementation of lightweight neural networks that provide service in real-time and keep the information private, such as surveillance systems that can detect and track people to keep it a safe area [71]. Likewise, the MistLearn framework [28], part of the MistGIS architecture, uses a deep neural network to run the *K-means* algorithm in low-power computers to classify medical profiles and diagnose diabetes. In order to set an artificial intelligence system effectively in motion on extreme edge devices, a good choice is to delegate the training phase to the Cloud, due to higher computing power, and to obtain a compressed version of the model which might not be as accurate to deploy in constrained devices [72].

In this spirit, a prominent methodology is combining data from different systems and intelligent entities to achieve higher accuracy in the detection of an event. Those multimodal data processing schemes are suitable for deep learning settings since they may gather data from multiple sources [73]. For instance, fall detection of elderly people is critical and must be reported immediately. Then, deploying a compressed convolutional neural network, which has been previously trained in the Cloud, inside an edge camera can contribute to efficiently make a decision [32]. This fusion of data can be undertaken at different levels of the architectural hierarchy to satisfy time constraints, privacy requirements, and heterogeneous data sources [31].

In [74], a MapReduce framework for edge devices based on a mobile agent is introduced. It uses three types of agents (Mapper, Worker, and Reducer), which can dynamically change the profile and move from the physical platform. It benefits from distributed architecture to store and process data using key-value patterns and to assign the running tasks to the devices

This computing power in the extreme edge results in convenient architectures to tackle Big Data problems because massive amounts of data generated at that level are reduced to save bandwidth, storage space, and energy, and we can decide which information has to be forwarded to the Fog or Cloud. Common approaches to improve the quality of information are local filtering and data fusion from available sources, taking advantage of the distributed essence. Besides that, this processing is prominently cooperative among the Mist devices, since they take advantage of traditional processing and machine learning techniques to guess multimodal events and operate on suited actuators.

Security

Last but not least, the implementation of novel computing paradigms always implies new security challenges that have to be imperatively faced in order to protect against known and unknown threats. This situation is compounded by the root property of Mist devices of operating with constrained resources, which may imply redesigning some authentication and confidentiality protocols, just for solving new specific problems.

The emergence of new use cases related to IoT brings difficulties to manage data along the network and control the access to it. Some of these issues are indicated in [75] using the privacy patterns concept to illustrate them in the smart vehicle scenario as an example, thus software engineers are aware of them and can find better designs.

Regarding the cryptography algorithms, they usually need powerful hardware to achieve good performance and constrained devices may encounter difficulties to satisfy this. In [76], a very comprehensive evaluation of RSA and ECDSA signature algorithms for TLS implementation is performed in microcontrollers with a dedicated hardware-acceleration module and comparing different levels of security. The results showed that ECC techniques are more efficient in terms of performance and energy consumption, and it demonstrated that alternative cryptography mechanisms are needed for SoCs.

Despite the lack of contributions about security in this field, the MistGIS architecture has included a framework, which satisfies the principles of confidentiality, integrity, and availability, based on a combination of Secure Sockets Layer (SSL) stack and role assignment to control the access to desired geographic information system (GIS) data [26].

Besides that, blockchain technology and related ones show up as good ways to secure IoT scenarios due to its distributed nature. In [77] a complete architecture for Information Transportation Systems (ITS), which applies blockchain smart contracts for registration and authentication of nodes, is described and, in this way, these actions of identity management can be done at any stage of the network.

The lowest tier of IoT architectures is still too limited in terms of cybersecurity because it was initially conceived to run fairly straightforward applications, and they are based on traditional methodologies, but it has evolved to provide a solution to sophisticated use cases which handle confidential data. Therefore, there must be more focus on efficient cryptography algorithms, secure network protocols, and privacy policies customized to collaborative networks of constrained devices, and it can rely on blockchain technology to make this paradigm shift a reality, in so doing, we can safeguard the inviolability of the system from unwelcome third parties. Notwithstanding the above, it became clear the huge importance of the Mist to change the way to preserve privacy and offshore data.

Critical comparison of distributed computing paradigms

Distributed computing paradigms have changed over the years according to the trends and demands of industry and academia. Currently, IoT sets the pace, and the Cloud has extended below to the Fog and the Edge until the Mist. Therefore, it is important to know the contribution of the Cloud and the others post-Cloud paradigms inside the IoT world. Additionally, with the discussion of the previous section, we have a better appreciation of the Mist layer and what new technological issues are involved, so its scope can be well-defined. Several aspects of each architectural layer are consequently examined to compare them and later summarized in Table 2.

While the **network architecture** of the Cloud can be regarded as centralized computing resources that transparently adapts itself to the demand, as we approach the edge of the network, decentralization of services increases, as well as vertical and horizontal cooperation and scalability are matters of concern [78, 79]. This change starts at the Fog level, where nodes can operate both standalone or in a federated cluster, and it fully explodes at the Edge and Mist layers where lightweight processes must be coordinated to offer a complete service regarding functional and non-functional requirements. This is also reflected in the chosen **service architecture**, which evolved from simple Client-Server flow in the Cloud, to disjointed models as SOA and Microservices [80], or even a very lightweight version called Nano-services [46].

The offshoring movement of resources implies a geographical aspect (**geolocation**) that is taken into account to satisfy real-time and security needs, since Cloud is usually located at any part of the Internet, but Fog, Edge and Mist are placed in specific environments which allow

Table 2 Essential features of Cloud, Fog, Edge and Mist Computing paradigms

	Cloud	Fog	Edge	Mist
Network Architecture	Centralized	Centralized/Decentralized	Decentralized	Decentralized
Service Architecture	Client-Server	SOA/Microservices	Microservices	Nano-services
Geolocation	Internet	Company/Institution	Network Border	Local Network
Awareness	No	Maybe	Yes	Yes
Mobility	Limited	Supported	Highly Supported	Highly Supported
Number Of Nodes	Tens	Tens/Hundreds	Tens/Hundreds	Hundreds/Thousands/Millions
Heterogeneity	Low	Medium	High	Very High
Communication	Wired Internet	Wired Intranet	Wired/Wireless Intranet	Wireless Ad-hoc Networks
Failure Risk	Medium	Low	Low	Very Low
Latency	High	Medium/Low	Low	Very Low
Network Core Usage	High	Low	Low	Low
Bandwidth	Medium	High	High	Medium
Data Privacy	Third Party	Institutional Access	Institutional Access	Local Access
Security	Open To All Users	Restricted Access	Distributed And Restricted Access	Distributed And Exclusively Local

to strictly run particular services [1]. Thus, resources in a closer or local network allow offering better quality services without any interruption and to know neighbor nodes to manage access control to data and to support elastic load changes.

Being aware of the localization of the devices and the context of operation (**awareness**) of each running service is essential in lower layers in order to enable location-based services and to ensure the lowest latency between proximity nodes [81]. In this way, new use cases that aim to revolutionize the world via distributed context and location awareness intelligence are only possible with post-Cloud paradigms deployments [1].

With respect to the above, another point to note is **mobility**, Cloud services are provided globally, but the ones at the edge are generally focused on a particular region to improve the quality of communications, that is, the Fog and the Edge tracks devices using a close connection, while the Mist automatically takes decisions about the local network services. In addition, it is necessary to build on a protocol that detaches identification from localization to make applications work uninterruptedly despite movement of final devices and allow the network provider to place applications at the best location dynamically [1, 81–83].

Along an entire post-Cloud infrastructure, big differences exist between the **number of nodes** that have to be managed by the operator at each architectural level. In the Cloud level, just a few servers are powerful enough to meet the needs of final users. However, in the Fog and Edge, nodes are associated to a specific location, then thousands of different networks must be launched to cover all regions, while in the extreme Edge millions of devices can be operating on the Internet of Things [81]. This marked increase in the number of active nodes also suffers from the heterogeneous nature of devices (**heterogeneity**), that is, devices at the bottom layers are characterized by constrained resources, multiple architecture designs, and uncertain network connection, thus the effort to administer them is higher [84]. To address these concerns, new virtualization techniques, which have less impact than ordinary virtual machines, have been developed, as is the case of containers in Fog/Edge Computing, or even very ad hoc lightweight solutions for the Mist [85].

On the **communication** side, Clouds are hard-wired on the Internet using standard technologies, while the Fog and Edge have also adopted modern mechanisms that allow to cope with ubiquitous challenges, and Mist networks are typically deployed in wireless environments with diverse connection ways [84]. Nonetheless, the fact of having several available communication paths and being able to run the application in the local environment makes services at the edge tolerant to hazardous failures and the availability of services is not affected for the proper functioning of operations [81]. So, reliability and robustness problems (**failure risk**) that may happen in the Cloud are usually fixed by moving concrete tasks near the point where data is originated.

Moving computing capabilities outside the Cloud has been motivated by emerging IoT applications that need imperatively better performance of communication systems to transform particular real-time use cases into reality, or even seize the new architecture designs to exploit features that otherwise would be impossible to have. As for the QoS, big differences among layers from top to bottom exist. The Cloud was initially used for monitoring and deferred data analytics, because the **latency** is extremely high, and nothing can be guaranteed in the communication path. Then the Fog tries to overcome this problem by approaching the servers to the end-devices and creating private slices that can be reserved to critical services. The final step has been taken by Edge and Mist Computing in bringing computing services to the extreme border, either directly connected to sensors and actuators or in the surrounding neighborhood. This may result in a very short delay in response time and the reduction in the **network core usage**, which could be saturated in terms of **bandwidth** [1, 81]. This last matter has also changed in trend in terms of **data privacy**, as only indispensable information needs to be shared with external agents [75].

A very important aspect not to be forgotten is **security** throughout the entire infrastructure. Current Cloud deployments are broadly protected against cyberattacks and most of the technologies used are widely known in the industry, and the community is usually aware of new threats. However, Cloud services are available to many different customers, so they must control many sources of danger. This also happens in Fog scenarios, but these resources are restricted to certain users. Therefore, the risk is lower. In the case of Edge and Mist layers, on the one hand, their decentralized nature makes systems more secure, but on the other hand, trust in devices may be difficult to ensure due to limited performance and random mobility patterns [86].

According to the examined features, we can now have better insight into the range of action of each cloudy computing paradigm. Due to the inherent remoteness of Cloud Computing, it deals with worldwide applications which offer no warranty for the users in terms of latency and provides unlimited computing power and storage capacity for future data processing of massive blocks of data. Owing to this lack of good quality of service for the final user, services at the Edge try to solve this concern by bringing computation over data sources and acting on it according to operating policies. In the specific case of Fog Computing, micro data centers are deployed to simulate Cloud functions for IoT scenarios where location aspects are vital. Otherwise, Mist Computing corresponds to taking full advantage of sensors and actuators in the local environment, forming an autonomous system that can run an IoT service using surrounding collaboration. It is also worth mentioning that Edge deployments mark a significant increase in terms of cost since they need distributed collaboration, architectural structures are more complex, and more physical resources are used.

As can be noted, the Mist has unlocked the possibility to exploit the capabilities of devices that take action on the environment, which are mainly resource-constrained and of varying kinds, like microcomputers, microcontrollers, multi-radio modules, powered by batteries, etc. Moreover, due to the wireless essence of the forming networks, heterogeneity is also present in the topology, from simple small star size networks to complex large multi-hop ones. All of this is carried out by means of giving smart capacities according to their resources or offloading specific tasks to surrounding nodes.

To sum up, we can highlight the fundamental points that make Mist Computing a revolutionary paradigm and make the difference between the Fog and the Edge. The Mist is composed of the *things* from the Internet of Things, in a manner that data is processed at the furthest reaches of the edge network where it is generated by sensors and actuators, taking heterogeneous WSN to the next level to be interoperable and cooperate in real-time. The growing connectivity era, either in short or wide area networks, enables smart distributed computing between Mist domains to empower collaborative intelligence in advanced multimodal settings where the locality of information is essential. This way, Mist systems are characterized by their autonomy to manage themselves without human interfaces, even in an isolated scenario where there is no Internet access. Additionally, the Mist contributes towards the enhancement of network conditions for critical applications, it operates and reacts to the environment faster, achieving extremely low latency and reducing the bandwidth consumed in the core of the network drastically. This also implies raising privacy control of the data, because it remains in the local network and does not rest on external services in the Cloud, Fog or Edge. A further advantage is advancement in mobility functionalities, since services are oriented to the data itself and its localized analytics, rather than the infrastructure.

Open challenges

With the advent of Mist Computing, a set of innovative challenges and opportunities comes up to develop a suite of technologies to improve existing solutions, which are not always suitable for this environment. Therefore, fundamental characteristics of the Mist, such as constrained resources, multi-network connectivity, extremely low latency, large volumes of data, and ubiquity of information, must be faced to make this paradigm evolve and be embraced by the industry. In this section, all the research lines that have been identified during review work and further reflection are described.

- New applications: enabling smart computing processing in devices directly allows developing and address hot application fields in IoT. An interesting case to look at is that of mobile ad-hoc networks (MANETs), which is growing lately, customized to ground vehicles (VANETs) and aerial vehicles (FANETs). Some examples are autonomous driving, which requires low latency communication in a multi-modal environment [4], or rescue and surveillance activities with UAVs, which are being applied to emergency situations where cooperative sensing is needed, and human presence is not recommended due to risk of danger [87]. Even smart agriculture, which relies on flying edge machines to enable cloudy mobile services [88] and military operations that require high performing real-time processing helped by machine learning algorithms at the edge [89].

- Architecture definition: the current technological ecosystem in the extreme edge is composed of many particular-case proposals, and there is no link between them. Then, as the OpenFog Consortium with Fog Computing, Mist Computing needs an association of companies, academic institutions, and sponsors that generate knowledge, standardize it, and promote it in flourishing use cases. This means defining a set of components and roles that must be implemented in Mist deployments, how they must communicate, and which technologies have to be supported in the standard.
- Flexible computing: capabilities of end devices, such as microcontrollers and microcomputers, have significantly improved during the past decade, now they can offer from lightweight parallelism to local GPU acceleration, including also virtualization mechanisms to isolate resources in multipurpose scenarios. Although the Mist is intended to provide processing autonomy to sensors and actuators, some tasks must be delegated to upper layers of network infrastructure to attain the desired results while ensuring power efficiency and low cost. Thus, Mist-oriented applications must support adaptive offloading in critical situations, looking for a balance between computing and communication power consumption, and combining mobility and replication in the system to offer continuous operation according to the computing and quality requirements, and leverage Fluid Computing paradigm.
- Unify IoT and 5G: the new generation of IoT has taken off at the same time 5G networks have come into existence. Indeed, both fields are closely related, and many synergies can be found since 5G networks are particularly oriented to advanced IoT applications, that is, both are aimed at reducing latency response and mass deployment of simultaneous devices. Specifically, the objective would be to use 5G infrastructure to manage and interconnect IoT devices. For instance, to configure orchestration actors to talk to *things* in the Mist to compose, execute, and dynamically adapt services on them, to apply new radio modulations to set wireless mesh networks using up-, down- and side-links in crowded settings, or even offloading intensive processing in MEC servers keeping QoS. In this way, there would be a complete Fluid Computing platform that allows to use all resources according to the application demand.
- Collaborative data analysis: the distributed nature of Mist Computing together with the massive deployment of sensors create the perfect combination to exploit efficient and scalable intelligent systems. A clear example of this is federated learning in the Edge, which trains independent tiny models in the local networks in order to aggregate them later and produce an upgraded version that can be distributed efficiently along the entire network [90]. It goes one step further with completely distributed data mining systems where edge nodes share their individual trained models to improve setup and performance and preserving the privacy of the raw data [91].
- Dataflow programming: popular service architectures, such as SOA and Microservices, are usually too heavy for limited-resource nodes and lack flexibility in mobility aspects. Dataflow models are regarded as good options to enable dynamic processing in a scenario where data is continuously generated and needs to be manipulated by different elements in separate locations. In this regard, in [92] a dataflow based on Zenoh is proposed as an encouraging mechanism that benefits from Pub/Sub messaging to compose a complete serverless mobile service [93].
- Resource allocation: in order to get the most of the resources from the *things* and facilitate the development of distributed applications in such a heterogeneous environment, mapping between actual physical resources and requested ones of the running application. For that purpose, virtual network embedding (VNE) techniques are used to automate this process and optimize the assignment in each setting. New algorithms must be suggested that fit the special features of constrained devices and guarantee good quality of service in smart perception networks [94].
- Privacy: Mist Computing gives the opportunity to change the present model of sharing and storing data according to privacy policies, which was established by the Cloud, i.e., gathering all information in an external centralized organization. Specifically, while using the Mist paradigm, most of the data processing operations are carried out in a controlled environ-

ment. Nevertheless, in some instances they have to be offloaded outside, so it is necessary to define new rules and patterns that must be followed to decide which agents can access which kind of data and how information must be shared, as well as distributed storage infrastructures that guarantee trustworthiness on data access. Similarly, these privacy principles can be applied to task management preserving authentication, confidentiality, and integrity using efficient cryptography algorithms [95].

- Blockchain: decentralization is a key feature in Mist Computing, but it results in challenging security administration. With the goal of overcoming this problem, Blockchain technology is regarded as the answer to register, authenticate and authorize multiple IoT devices using a distributed ledge composed of efficient smart contracts in each device [77]. This way, response time at the edge is improved, resource utilization is maximized, and reliability is enhanced, achieving a more robust system. Energy-aware developments: like any sensor network, Mist devices usually draw on wireless communications powered by batteries, in a way that they must be supplied permanently to guarantee that they will operate for a long lifetime without replacement. In order to achieve this, sophisticated energy harvesting techniques must be implemented, but we must also focus on developing low power computation that may be aware of the dynamic power source level. Both power saving communication protocols, and data dissemination and analysis, such as lightweight machine learning, must be precisely combined with new alternative energy sources [96] to run complex systems in the Mist.

These are some functional and non-functional challenges that pose interesting research studies and objectives for the near future, and that may contribute to designing an interoperable framework to serve as middleware platform in Mist scenarios.

Conclusion

All this work is intended to underscore the importance of a new post-Cloud computing paradigm, such as Mist Computing, in the development of distributed intelligent applications in new use cases, which are trending currently, and its impact on the Internet of Things to make perception systems smarter. In fact, the Mist gives the chance to exploit resources at the extreme edge of the network, basically sensors and actuators, in a distributed and collaborative manner maintaining service operations to respond to network changes and taking care of data privacy.

To fully achieve this task, an Integrative Literature Review has been conducted, since it restricts the examined proposals and helps to guide the work cogently. Notwithstanding the rigidity of the selected process, the quality of the outcome is right, and it covers the vast majority of this emerging topic. This general state-of-the-art review provides a clear understanding of this new and growing theme and pinpoints the fields where considerable contributions have been made.

Further to this contribution, the paper makes a consistent analysis of the characteristics of architectural layers from up, the Cloud, to down, the Mist, in order to aid the reader to determine the extent of each network model. It also goes over work lines that need to be addressed in the Mist layer in accordance with its natural structure and have been identified throughout the drafting of the article.

With a view to a near term, the creation of a distributed nano-service platform, which allows deploying specific functionalities in devices and compose them, is envisioned to be the next step in Mist Computing. To do this, modern communication patterns are going to be used to give appropriate dynamic connectivity to realign the solutions with future Fluid Computing.

In conclusion, the pertinence of this new computing model towards challenging present-day use cases is shown throughout the article. In particular, the core idea of Mist Computing referred to on-demand collaborative distributed systems, can be successfully applied in intensive Big Data context, multi-modal decision-making mechanism or critical real-time location-awareness situations.

Abbreviations

API: Abstract Programming Interface; CDN: Content Delivery Network; CIoT: Customer Internet of Things; DAG: Direct Acyclic Graph; DDF: Distributed Dataflow; DDS: Data Distribution Service; DHT: Distributed Hash Table; ESB: Enterprise Service Bus; FANET: Flying Ad-hoc Network; FSD: Fair Storage Distribution; GIS: Geographic Information System; IIoT: Industrial Internet of Things; IFoT: Information Flow of Things; IoHT: Internet of Health Things; IoT: Internet of Things; ITS: Information Transportation Systems; MANET: Mobile Ad-hoc Network; MEC: Mobile or Multi-access Edge Computing; MIoT: Massive Internet of Things; MQTT: Message Queuing Telemetry Transport; RMN: Regular Mist Node; SMN: Super Mist Node; SOA: Service-oriented Architecture; SSL: Secure Sockets Layer; VANET: Vehicle Ad-hoc Network; vFog: virtual Fog; VNE: Virtual Network Embedding; WSN: Wireless Sensor Network.

Acknowledgements

This work is supported by the DocTIC PhD program of the University of Vigo under the promotion of Formación de Profesorado Universitario (FPU, FPU19/01284) fellowship by the Spanish Ministry of Science, Innovation and Universities.

Authors' contributions

Conceptualization, Methodology and Validation of the manuscript have been directed by JJLE, RPDR and FGC. Data Curation, Formal Analysis and Writing - Original Draft Writing have been conducted by JJLE, and Writing - Review & Editing has been done by JJLE, RPDR and FGC. Supervision and Funding

Acquisition have been carried out by RPDR and FGC. All authors have read and agreed to the published version of the manuscript.

Authors' information

Juan José López Escobar is a Telecommunications Engineer from University of Vigo (BSc and MSc), where he is currently enrolled in the PhD in Information and Communications Technology and is also a researcher at the department of Telematics Engineering in the Information Technology Group (GTI). His main research areas are wireless protocols and technologies for the Internet of Things and cellular communications, and distributed computing paradigms (Cloud, Fog, Edge and Mist). The doctoral stage has been funded by a grant for university professor training (FPU) from the Spanish Ministry of Science, Innovation and Universities, which allows developing both academic and research skills.

Rebeca P. Díaz Redondo is currently an Associate Professor with the Department of Telematics Engineering, University of Vigo. She is currently working on defining appropriate architectures for distributed and collaborative data analysis, especially thought for IoT solutions, where computation must be on the edge of the network. Rebeca has participated in more than 40 projects and 25 works of technological transfer through contracts with companies and/or public institutions. She is currently involved in the scientific and technical activities of several National and European research & educative projects.

Felipe Gil-Castiñeira is currently an Associate Professor with the Department of Telematics Engineering, University of Vigo. His interests include wireless communication and core network technologies, multimedia communications, embedded systems, ubiquitous computing and the Internet of things. He has published over sixty papers in international journals and conference proceedings. He has led several national and international R &D projects. He holds two patents in mobile communications.

Funding

This research has received funding from the European Union's Horizon 2020 research and innovation program ECSEL Joint Undertaking (JU) under grant agreement No. 876487, NextPerception project "Next Generation Smart Perception Sensors and Distributed Intelligence for Proactive Human Monitoring in Health, Wellbeing, and Automotive Systems", cofunded by the Spanish Ministry of Science and Innovation (PCI2020-112174). It has also been supported by the European Regional Development Fund (ERDF) and the Galician Regional Government, under the agreement for funding the Atlanttic Research Center for Information and Communication Technologies (AtlantTIC), and the Spanish Government under re-search project "Enhancing Communication Protocols with Machine Learning while Protecting Sensitive Data (COMPROMISE)" (PID2020-113795RB-C33/AEI/10.13039/501100011033).

Availability of data and materials

Not applicable.

Declarations

Competing interests

The authors declare that they have no competing interests.

Author details

[1]Information Technologies Group (GTI), atlanTTic Research Center, University of Vigo, 36210 Vigo, Spain. [2]Information & Computing Lab (I&CLab), atlanTTic Research Center, University of Vigo, 36210 Vigo, Spain.

Received: 31 January 2022 Accepted: 26 October 2022
Published online: 19 November 2022

References

1. Yousefpour A, Fung C, Nguyen T, Kadiyala K, Jalali F, Niakanlahiji A, Kong J, Jue JP (2019) All one needs to know about fog computing and related edge computing paradigms: A complete survey. J Syst Archit 98:289–330. https://www.sciencedirect.com/science/article/pii/S1383762118306349
2. Bhattacharya S, Senapati S, Soy SK, Misra C, Barik R (2020) Performance analysis of enhanced mist-assisted cloud computing model for healthcare system. In: 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA). IEEE, pp 1–5
3. Galambos P (2020) Cloud, fog, and mist computing: Advanced robot applications. IEEE Syst Man Cybernet Mag 6(1):41–45
4. Bonanni M, Chiti F, Fantacci R (2020) Mobile mist computing for the internet of vehicles. Internet Technol Lett 3(6):e176
5. Paci F, Brunelli D, Benini L (2018) Lightweight io virtualization on mpu enabled microcontrollers. ACM SIGBED Rev 15(1):50–56
6. Sánchez-Gallegos DD, Galaviz-Mosqueda A, Gonzalez-Compean J, Villarreal-Reyes S, Perez-Ramos AE, Carrizales-Espinoza D, Carretero J (2020) On the continuous processing of health data in edge-fog-cloud computing by using micro/nanoservice composition. IEEE Access 8:120255–120281
7. Alam F, Mehmood R, Katib I, Albogami NN, Albeshri A (2017) Data fusion and iot for smart ubiquitous environments: A survey. IEEE Access 5:9533–9554
8. Wang Y, Thulasiraman P (2020) Post-cloud computing models and their comparisons. In: International Conference on Cloud Computing. Springer, Cham, pp 141–151
9. You X, Wang CX, Huang J, Gao X, Zhang Z, Wang M, Huang Y, Zhang C, Jiang Y, Wang J et al (2021) Towards 6g wireless communication networks: Vision, enabling technologies, and new paradigm shifts. Scie China Inf Sci 64(1):1–74
10. Harjula E, Artemenko A, Forsström S (2021) Edge computing for industrial iot: challenges and solutions. Wireless Networks and Industrial IoT. Springer, Cham, pp 225–240
11. Bonomi F, Milito R, Zhu J, Addepalli S (2012) Fog computing and its role in the internet of things. Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing (MCC '12). Association for Computing Machinery, New York, pp 13–16
12. Vaquero LM, Rodero-Merino L (2014) Finding your way in the fog: Towards a comprehensive definition of fog computing. ACM SIGCOMM Comput Commun Rev 44(5):27–32
13. Satyanarayanan M (2017) The emergence of edge computing. Computer 50(1):30–39
14. Chen CH, Lin MY, Liu CC (2018) Edge computing gateway of the industrial internet of things using multiple collaborative microcontrollers. IEEE Netw 32(1):24–32
15. Pang Z, Sun L, Wang Z, Tian E, Yang S (2015) A survey of cloudlet based mobile computing. In: 2015 International Conference on Cloud Computing and Big Data (CCBD). IEEE, pp 268–275
16. Iorga M, Feldman L, Barton R, Martin M, Goren N, Mahmoudi C (2018) Fog computing conceptual model. Special Publication (NIST SP), National Institute of Standards and Technology, Gaithersburg
17. Wang Y (2015) Cloud-dew architecture. Int J Cloud Comput 4(3):199–210
18. Ray PP (2017) An introduction to dew computing: definition, concept and implications. IEEE Access 6:723–737
19. Corsaro A (2016) Fluid computing: Unifying cloud, fog, and mist computing. https://www.embeddedcomputing.com/application/misc/fluid-computing-unifying-cloud-fog-and-mist-computing. Accessed 20 May 2021
20. Swain BR, Sahoo JJ, Prasad A, Selvam DT (2019) Rise of fluid computing: A collective effort of mist, fog and cloud. Int J Comput Sci Eng 7(4):62–69
21. Torraco RJ (2005) Writing integrative literature reviews: Guidelines and examples. Hum Resour Dev Rev 4(3):356–367
22. Dutta A, Misra C, Barik RK, Mishra S (2021) Enhancing mist assisted cloud computing toward secure and scalable architecture for smart healthcare. Advances in Communication and Computational Technology. Springer, Singapore, pp 1515–1526
23. Asif-Ur-Rahman M, Afsana F, Mahmud M, Kaiser MS, Ahmed MR, Kaiwartya O, James-Taylor A (2018) Toward a heterogeneous mist, fog, and cloud-based framework for the internet of healthcare things. IEEE Internet Things J 6(3):4049–4062
24. Shaikh TA, Ali R (2021) Fog Computing for Healthcare 4.0 Environments. Fog-iot environment in smart healthcare: A case study for student stress monitoring. Springer, Cham, pp 211–250
25. Preden JS, Tammemäe K, Jantsch A, Leier M, Riid A, Calis E (2015) The benefits of self-awareness and attention in fog and mist computing. Computer 48(7):37–45
26. Barik RK, Dubey AC, Tripathi A, Pratik T, Sasane S, Lenka RK, Dubey H, Mankodiya K, Kumar V (2018) Mist data: leveraging mist computing for

secure and scalable architecture for smart and connected health. Procedia Comput Sci 125:647–653
27. Barik RK, Tripathi A, Dubey H, Lenka RK, Pratik T, Sharma S, Mankodiya K, Kumar V, Das H (2018) Mistgis: Optimizing geospatial data analysis using mist computing. Progress in Computing, Analytics and Networking. Springer, Singapore, pp 733–742
28. Barik RK, Priyadarshini R, Dubey H, Kumar V, Yadav S (2018) Leveraging machine learning in mist computing telemonitoring system for diabetes prediction. Advances in Data and Information Sciences. Springer, Singapore, pp 95–104
29. El-Hasnony IM, Mostafa RR, Elhoseny M, Barakat SI (2020) Leveraging mist and fog for big data analytics in iot environment. Trans Emerg Telecommun Technol 32:(7):e4057
30. Pulli P, Martikainen O, Zhang Y, Naumov V, Asghar Z, Pitkänen A (2011) Augmented processes: A case study in healthcare. In: Proceedings of the 4th International Symposium on Applied Sciences in Biomedical and Communication Technologies. pp 1–6
31. Dautov R, Distefano S, Buyya R (2019) Hierarchical data fusion for smart healthcare. J Big Data 6(1):1–23
32. Divya V, Sri RL (2020) Intelligent real-time multimodal fall detection in fog infrastructure using ensemble learning. Challenges and Trends in Multimodal Fall Detection for Healthcare. Springer, Cham, pp 53–79
33. Oteafy SM, Hassanein HS (2018) Iot in the fog: A roadmap for data-centric iot development. IEEE Commun Mag 56(3):157–163
34. Patel P, Ali MI, Sheth A (2017) On using the intelligent edge for iot analytics. IEEE Intell Syst 32(5):64–69
35. Markakis EK, Karras K, Zotos N, Sideris A, Moysiadis T, Corsaro A, Alexiou G, Skianis C, Mastorakis G, Mavromoustakis CX et al (2017) Exegesis: Extreme edge resource harvesting for a virtualized fog environment. IEEE Commun Mag 55(7):173–179
36. Crăciunescu M, Chenaru O, Dobrescu R, Florea G, Mocanu Ş (2019) Iiot gateway for edge computing applications. In: International Workshop on Service Orientation in Holonic and Multi-Agent Manufacturing. Springer, Cham, pp 220–231
37. Battistoni P, Sebillo M, Vitiello G (2019) Experimenting with a fog-computing architecture for indoor navigation. In: 2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC). IEEE, pp 161–165
38. Santamaria AF, Raimondo P, Tropea M, De Rango F, Aiello C (2019) An iot surveillance system based on a decentralised architecture. Sensors 19(6):1469
39. Zyrianoff I, Heideker A, Silva D, Kleinschmidt J, Soininen JP, Salmon Cinotti T, Kamienski C (2020) Architecting and deploying iot smart applications: A performance-oriented approach. Sensors 20(1):84
40. Dogo EM, Salami AF, Aigbavboa CO, Nkonyana T (2019) Taking cloud computing to the extreme edge: A review of mist computing for smart cities and industry 4.0 in africa. Edge Comput 107–132
41. Tammemäe K, Jantsch A, Kuusik A, Preden JS, Õunapuu E (2018) Self-aware fog computing in private and secure spheres. Fog Computing in the Internet of Things. Springer, Cham, pp 71–99
42. Nakamura Y, Suwa H, Arakawa Y, Yamaguchi H, Yasumoto K (2016) Design and implementation of middleware for iot devices toward real-time flow processing. In: 2016 IEEE 36th International Conference on Distributed Computing Systems Workshops (ICDCSW). IEEE, pp 162–167
43. Talusan JP, Nakamura Y, Mizumoto T, Yasumoto K (2018) Near cloud: Low-cost low-power cloud implementation for rural area connectivity and data processing. In: 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), vol 2. IEEE, pp 622–627
44. Liyanage M, Chang C, Srirama SN (2016) mepaas: mobile-embedded platform as a service for distributing fog computing to edge nodes. In: 2016 17th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT). IEEE, pp 73–80
45. Small N, Akkermans S, Joosen W, Hughes D (2017) Niflheim: An end-to-end middleware for applications on a multi-tier iot infrastructure. In: 2017 IEEE 16th International Symposium on Network Computing and Applications (NCA). IEEE, pp 1–8
46. Harjula E, Karhula P, Islam J, Leppänen T, Manzoor A, Liyanage M, Chauhan J, Kumar T, Ahmad I, Ylianttila M (2019) Decentralized iot edge nanoservice architecture for future gadget-free computing. IEEE Access 7:119856–119872
47. Sattari A, Ehsani R, Leppänen T, Pirttikangas S, Riekki J (2020) Edge-supported microservice-based resource discovery for mist computing. In: 2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech). IEEE, pp 462–468
48. Pratik T, Lenka RK, Nayak GK, Kumar A (2018) An architecture to support interoperability in iot devices. In: 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN). IEEE, pp 705–710
49. Zhang L, Afanasyev A, Burke J, Jacobson V, Claffy K, Crowley P, Papadopoulos C, Wang L, Zhang B (2014) Named data networking. ACM SIGCOMM Comput Commun Rev 44(3):66–73
50. Amadeo M, Ruggeri G, Campolo C, Molinaro A, Loscrí V, Calafate CT (2019) Fog computing in iot smart environments via named data networking: A study on service orchestration mechanisms. Future Internet 11(11):222
51. Zhang D, Chan CC, Zhou GY (2018) Enabling industrial internet of things (iiot) towards an emerging smart energy system. Glob Energy interconnection 1(1):39–47
52. Sonbol K, Özkasap Ö, Al-Oqily I, Aloqaily M (2020) Edgekv: Decentralized, scalable, and consistent storage for the edge. J Parallel Distrib Comput 144:28–40
53. Persson P, Angelsmark O (2015) Calvin-merging cloud and iot. Procedia Comput Sci 52:210–217
54. Mehta A, Baddour R, Svensson F, Gustafsson H, Elmroth E (2017) Calvin constrained—a framework for iot applications in heterogeneous environments. In: 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS). IEEE, pp 1063–1073
55. Donassolo B, Fajjari I, Legrand A, Mertikopoulos P (2019) Fog based framework for iot service provisioning. In: 2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC). IEEE, pp 1–6
56. Liyanage M, Chang C, Srirama SN (2018) Adaptive mobile web server framework for mist computing in the internet of things. Int J Pervasive Comput Commun
57. Dautov R, Distefano S, Bruneo D, Longo F, Merlino G, Puliafito A (2021) Data agility through clustered edge computing and stream processing. Concurr Comput Pract Experience 33(7):1–1
58. Zanella M, Massari G, Galimberti A, Fornaciari W (2018) Back to the future: Resource management in post-cloud solutions. In: Proceedings of the Workshop on INTelligent Embedded Systems Architectures and Applications. pp 33–38
59. Hong K, Lillethun D, Ramachandran U, Ottenwälder B, Koldehofe B (2013) Mobile fog: A programming model for large-scale applications on the internet of things. In: Proceedings of the second ACM SIGCOMM workshop on Mobile cloud computing. pp 15–20
60. Saurez E, Hong K, Lillethun D, Ramachandran U, Ottenwälder B (2016) Incremental deployment and migration of geo-distributed situation awareness applications in the fog. In: Proceedings of the 10th ACM International Conference on Distributed and Event-based Systems. pp 258–269
61. Giang NK, Blackstock M, Lea R, Leung VC (2015) Developing iot applications in the fog: A distributed dataflow approach. In: 2015 5th International Conference on the Internet of Things (IOT). IEEE, pp 155–162
62. Cheng B, Solmaz G, Cirillo F, Kovacs E, Terasawa K, Kitazawa A (2017) Fogflow: Easy programming of iot services over cloud and edges for smart cities. IEEE Internet Things J 5(2):696–707
63. Corsaro A, Baldoni G (2018) fogø5: Unifying the computing, networking and storage fabrics end-to-end. In: 2018 3rd Cloudification of the Internet of Things (CIoT). IEEE, pp 1–8
64. Shahraki A, Geitle M, Haugen Ø (2020) A comparative node evaluation model for highly heterogeneous massive-scale internet of things-mist networks. Trans Emerg Telecommun Technol 31(12):e3924
65. Vasconcelos D, Andrade R, Severino V, Souza JD (2019) Cloud, fog, or mist in iot? that is the question. ACM Trans Internet Technol (TOIT) 19(2):1–20
66. Linaje M, Berrocal J, Galan-Benitez A (2019) Mist and edge storage: Fair storage distribution in sensor networks. IEEE Access 7:123860–123876
67. Grover J, Garimella RM (2018) Reliable and fault-tolerant iot-edge architecture. In: 2018 IEEE SENSORS. IEEE, pp 1–4
68. Rubio-Drosdov E, Sánchez DD, Almenárez F, Marín A (2019) A framework for efficient and scalable service offloading in the mist. In: 2019 IEEE 5th World Forum on Internet of Things (WF-IoT). IEEE, pp 460–463

69. Battistoni P, Sebillo M, Vitiello G (2019) Computation offloading with mqtt protocol on a fog-mist computing framework. In: International Conference on Internet and Distributed Computing Systems. Springer, Cham, pp 140–147
70. Mihai V, Hanganu CE, Stamatescu G, Popescu D (2018) Wsn and fog computing integration for intelligent data processing. In: 2018 10th International Conference on Electronics, Computers and Artificial Intelligence (ECAI). IEEE, pp 1–4
71. Xu R, Nikouei SY, Chen Y, Polunchenko A, Song S, Deng C, Faughnan TR (2018) Real-time human objects tracking for smart surveillance at the edge. In: 2018 IEEE International Conference on Communications (ICC). IEEE, pp 1–6
72. Debauche O, Mahmoudi S, Mahmoudi SA, Manneback P, Lebeau F (2020) A new edge architecture for ai-iot services deployment. Procedia Comput Sci 175:10–19
73. Ma Q, Nie Y, Song J, Zhang T (2020) Multimodal data processing framework for smart city: A positional-attention based deep learning approach. IEEE Access 8:215505–215515
74. Satoh I (2013) A framework for data processing at the edges of networks. In: International Conference on Database and Expert Systems Applications. Springer, Berlin, Heidelberg, pp 304–318
75. Pape S, Rannenberg K (2019) Applying privacy patterns to the internet of things'(iot) architecture. Mob Netw Appl 24(3):925–933
76. Suárez-Albela M, Fraga-Lamas P, Fernández-Caramés TM (2018) A practical evaluation on rsa and ecc-based cipher suites for iot high-security energy-efficient fog and mist computing devices. Sensors 18(11):3868
77. Sharma PK, Park JH (2020) Blockchain-based secure mist computing network architecture for intelligent transportation systems. IEEE Trans Intell Transp Syst
78. McCann J, Quinn L, McGrath S, O'Connell E (2018) Towards the distributed edge–an iot review. In: 2018 12th International Conference on Sensing Technology (ICST). IEEE, pp 263–268
79. Yeow K, Gani A, Ahmad RW, Rodrigues JJ, Ko K (2017) Decentralized consensus for edge-centric internet of things: A review, taxonomy, and research issues. IEEE Access 6:1513–1524
80. Butzin B, Golatowski F, Timmermann D (2016) Microservices approach for the internet of things. In: 2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA). IEEE, pp 1–6
81. Sunyaev A (2020) Fog and edge computing. Internet Computing: Principles of Distributed Systems and Emerging Internet-Based Technologies. Springer, Cham, pp 237–264
82. Yu W, Liang F, He X, Hatcher WG, Lu C, Lin J, Yang X (2017) A survey on the edge computing for the internet of things. IEEE Access 6:6900–6919
83. El-Sayed H, Sankar S, Prasad M, Puthal D, Gupta A, Mohanty M, Lin CT (2017) Edge of things: The big picture on the integration of edge, iot and the cloud in a distributed computing environment. IEEE Access 6:1706–1717
84. Portilla J, Mujica G, Lee JS, Riesgo T (2019) The extreme edge at the bottom of the internet of things: A review. IEEE Sens J 19(9):3179–3190
85. Mansouri Y, Babar MA (2021) A review of edge computing: Features and resource virtualization. J Parallel Distrib Comput
86. Zhang P, Zhou M, Fortino G (2018) Security and trust issues in fog computing: A survey. Futur Gener Comput Syst 88:16–27
87. Popescu D, Stoican F, Stamatescu G, Chenaru O, Ichim L (2019) A survey of collaborative uav-wsn systems for efficient monitoring. Sensors 19(21):4690
88. Uddin MA, Ayaz M, Mansour A, Sharif Z, Razzak I, et al (2021) Cloud-connected flying edge computing for smart agriculture. Peer-to-Peer Netw Appl 1–11
89. Shahid H, Shah MA, Almogren A, Khattak HA, Din IU, Kumar N, Maple C (2021) Machine learning-based mist computing enabled internet of battlefield things. ACM Trans Internet Technol (TOIT) 21(4):1–26
90. Mills J, Hu J, Min G (2019) Communication-efficient federated learning for wireless edge intelligence in iot. IEEE Internet Things J 7(7):5986–5994
91. Sun X, Xu R, Wu L, Guan Z (2021) A differentially private distributed data mining scheme with high efficiency for edge computing. J Cloud Comput 10(1):1–12
92. Baldoni G, Loudet J, Cominardi L, Corsaro A, He Y (2021) Facilitating distributed data-flow programming with eclipse zenoh: the erdos case. In: Proceedings of the 1st Workshop on Serverless mobile networking for 6G Communications. pp 13–18
93. Teranishi Y, Kimata T, Yamanaka H, Kawai E, Harai H (2017) Dynamic data flow processing in edge computing environments. In: 2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC), vol 1. IEEE, pp 935–944
94. Wu D, Liu Z, Yang Z, Zhang P, Wang R, Ma X (2021) Survivability-enhanced virtual network embedding strategy in virtualized wireless sensor networks. Sensors 21(1):218
95. Hosen AS, Sharma PK, Ra IH, Cho GH (2021) Sptm-ec: A security and privacy-preserving task management in edge computing for iiot. IEEE Trans Ind Inform
96. Jokic P, Emery S, Benini L (2021) Battery-less face recognition at the extreme edge. In: 2021 19th IEEE International New Circuits and Systems Conference (NEWCAS). IEEE, pp 1–4

Publisher's Note