Creating Intelligent Computational Edge through Semantic Mediations

Radmila Juric
ALMAIS Consultancy, UK
radjur3@gmail.com

Elisabetta Ronchieri
CNAF/INFN, Italy
elisabetta.ronchieri@cnaf.infn.it

Sang Suh
Texas A&M University Commerce
Sang.Suh@tamuc.edu

Abstract

This paper proposes semantic mediation based on reasoning and the first order logic for mediating the best possible configuration of computational edge, relevant for software applications which may benefit for running computations with proximity to their data sources. The mediation considers the context in which these applications exist and exploits the semantic of that context for decision making on where computational elements should reside and which data they should use. The application of semantic mediation could address the initiative to accommodate algorithms from predictive and learning technologies, push AI towards computational edges and potentially contribute towards creating a computing continuum.

Keywords: edge, computing, semantic, mediation.

1. Introduction

In the last five years, the Edge Computing Paradigm (ECP) has captured our attention and become a cornerstone of modern computational capabilities (Satyanarayanan, 2017), (Lopez et al., 2015), (CISCO, 2010). For a rather long time, we have had problems with the dominance of cloud computing (Zhang et al., 2015), Taleb et al., 2019) and started thinking about a new computational paradigm which addresses demands of modern computing and pitfalls of clouds. This applies to many software applications but is prevalent in the domain of the Internet of Things (IoT). It is characterized by the existence of numerous interconnected devices, with constantly increasing computational power, possibly dependent on sensory technologies and often in charge of humans, who depend on mobile wireless networks and mobile computing in their everyday lives. It is no surprise that the decentralization of clouds, triggered by location awareness, was promoted through fog computing (Vaquero et al., 2014), (Mukherjee, 2018) and cloudlets (Verbelen et al., 2012) (Satyanarayanan, 2014), which had instant appeal for the IoT and their applications. It was followed by the coexistence of edge/fog/clouds (Seal and Mukherjee, 2018), which outlined promises of edge computing (Shi and Dusdar, 2016) despite its challenges ((Pisani and Borin, 2018)). Fog and edge computing are seen as solutions in creating mobile and wireless computing and accommodating pervasiveness and the IoT in particular (Angel et al., 2022).

Mobility in computing is a sine qua non in the 21st century. We focus on its taxonomies and challenges, examine the future of Mobile Edge Computing (MEC) (Abbas et al., 2018), and open edge computing for ubiquitous Artificial Intelligence (AI) (Mahdavinejad, 2018), (Deng et al., 2020), and create intelligent computational edge. We need a commonly accepted ECP, supported by computational models, software technologies, infrastructures and advances in communications and networking. We expect a realistic computational power at edges.

In the light of advances in computing with predictive and learning technologies, which shape current AI, it is expected that we can squeeze deep learning into mobile devices (Lane et al., 2017) or even talk about edge analytics in the IoT (Merenda et al., 2020). Advances in software, communication technologies and the proliferation of computational devices around us impose demands on the ECP and trigger questions we may not be able to answer. Can the world of pervasive computing, which comes close to the edges of computer networks, become intelligent? Could we transfer advances in AI into the computing edge and create intelligence? What would the role of clouds be with intelligent computational edge? Could we learn something from the IoT field and apply it to the ECP?

Interesting forward thinking from (Rausch and Dustdar, 2019) promotes convergence of humans, things, and the AI to create edge intelligence. It was triggered by research on human augmentations or augmenting human abilities through technologies (Raisano et al., 2019), (Saracco, 2018) or looking at the IoT for augmenting humans (Pimagomedov, 2021). The idea of involving humans in managing *localized intelligence* in ECP, has not been widely adopted, although computational edge within the IoT and cyber physical spaces are often in the hands of humans and software as a service is tailored mostly for humans.

There is one problem common to all claims of seeing a bright future for intelligent edge computing.

URI: https://hdl.handle.net/10125/103456
978-0-9981331-6-4

Currently we do not consider creating computational intelligence without predictive/learning technologies and machine learning (ML) in particular. We thrive in fast, cheap, and sometimes efficient processing of a vast amount of data and claim that we create intelligence through predictive inference.

Would this be the only way forward in mastering machine intelligence? Shall we ever achieve computational intelligence suitable for the ECP, if we stick to algorithms of current AI and never consider anything else?

In this paper we explore possibilities of creating intelligent computational edge by making it independent in terms of deciding *which computations we wish to run at the edge and which data the edge needs*. These decisions are constantly changeable, and dependent on circumstances, or contexts in which we empower the edge. It does not mean that the ties with clouds are cut, but if we perform reasoning upon the semantics stored in the context where edge computing performs computational tasks, then we could mediate about the best possible option on *where computing happens and why*. The space between computing edges and clouds is vast and we should exploit it by taking advantage of potentially powerful edge, addressing shortcoming of cloud computing at the same time, and mediating on coexistence of edge/fog/clouds.

Will these ideas enable a computing continuum? Is a computing continuum a possible outcome in creating intelligent computational edge? Would edge computing become essential in human or machine augmentation? If we add mediation based on reasoning to the current concept of running AI, as close to mobile data source as possible, would we experience a real power of the ECP?

The paper is organized as follows. Section 2 formulates the problem through publications on combinations of resources and services between edges and clouds. Section 3 gives a taxonomical model for semantic mediation, which enables reasoning upon the best possible edge computing configuration. Section 4 proposes a reasoning process for mediation based on SWRL enabled OWL ontologies. The scenario used for mediating on the types of ML algorithms and data needed for them for configuring computational edge, is placed in the public health sector domain, during an epidemic. Section 5 overviews related work and in Section 6 gives conclusions.

2. Formulating the Problem

One of the first ideas of seamlessly combining resources and services on a pathway from the edge to clouds appeared in 2019 (Hao et al., 2020). The authors compare their ideas to the fluid ecosystem in which services are federated and orchestrated on demand, with provisions for real time reaction to unexpected situations in data processing. Orchestration and federation also have roots in the late 90s. They addressed interoperability in heterogeneous databases. This is all still applicable, particularly in data driven applications and their proposal can be seen as an example of *computing in a continuum*.

In (Balouek et al., 2019) the term *computing continuum* is reserved for edge to cloud integration for data driven workflows. Edge resources are still seen as not capable to support any data intensive computing, and therefore we must rely on cloud services. However, the paper also proposes the use of federation for infrastructure and programming services, which create dynamic workflows to deal with various unexpected situations in a computing continuum.

The proposal from (Milojcic, 2020) also uses the term *ecosystem* in relation to a computing continuum, because its aim is to seamlessly combine resources and services at the cloud and edges, plus *in-transit*, along the data path from the edge to clouds. In (Satyanarayann and Davis, 2019), the cloud-edge continuum works on the principle of the orchestration of operations between edges and clouds, where context awareness helps in deciding where to store data and perform computations.

All these publications are rather new and there are no follow-ups, but there are new paradigms promoted, such as a cognitive computing continuum (Carvalho, 2020)). It gives the provision of services for dynamic IoTs and "*distributed, opportunistic, self-managed collaboration of devices within the IoT*", which may happen to reside outside the data centers and clouds.

Note: the word *cognition* is not related to cognitive computing. It applies to handheld devices in the proximity of data sources, which make autonomous decisions, based on the data sensed from their environments. None of them address the existence of the ECP and there is space for thinking differently and looking at mediation for configuring edge computing.

2.1. What Would our Vision Be?

First, we focus on semantic mediation based on reasoning and first order logic for mediating the best possible configuration of computational edge. Second, the mediation considers the context in which an instance of software applications exists, like instances of the IoT. It exploits the semantic of that context for decision making on *where computational elements should reside and which data they should use*. Third, the creation of a computing continuum may be feasible, because it may accommodate algorithms from predictive and learning technologies at the computational edge and spread it in the space between the edge and clouds if required.

The idea of mediation in computer science (Wiederhold, 1995) is rather old, as are federation and orchestration. It was exploited 30 years ago, as a predecessor of client server technologies and middleware architectures, for resolving heterogeneities in software applications and platforms and addressing semantic interoperability in databases, developed through the 80s and 90 (Juric et al., 2004). However, the mediation was short lived for one reason. Client server computing took off and enabled service-oriented software architectures which addressed the problem of heterogeneity to a certain extent (Juric et al., 2004a). 30 years ago, we did not have a computational model which could support mediations. Without logic inference and reasoning, it was impossible to mediate about anything and have efficient computing without a huge software application overhead. It was only after adopting semantic web technologies (SWT) that we could compute with logic reasoning. It has been used in the semantic web since then. The SWT and languages, run efficient reasoning upon abstract and computable concepts (Shojanoori, 2013), (Kataria, 2011).

If we merge the idea of mediation with logic inference, we could address numerous problems in configurations of computational edge, the existence of instances of the IoT and address the problem of creating an infrastructure for a computational continuum. Would this be the way forward for creating an ECP?

The interest of merging the two for creating an ECP is triggered by successes in using the SWT for creating computational models, powered by SWRL enabled OWL ontologies (Juric, 2016) (Juric and Kim, 2017), (Shojanoori and Juric, 2014).. At this moment there is nothing else on offer in either computer science or communication theories or predictive technologies or engineering which could take us forward to a fully functional ECP. This is particularly true in cases of instances of the IoT. Therefore, it is worthwhile looking at semantic mediation as an answer to creating an efficient computational edge and possibly continuum.

3. The Proposal for Semantic Mediation

Figure 1 shows a conceptual model where reasoning based on SWT is used upon SWRL enabled OWL ontologies (WWW Layer Cake, 2004), (W3C OWL group, 2004) for mediating the best possible configuration of cyber physical spaces, the IoT and edge computing, which may utilize computational continuum and secure the running of current AI algorithms.

The mediation itself would be represented by logic reasoning using SWRL, but there is something more important to say first. The left part of the Fig. 1 contains the semantic for creating ontological concepts to enable reasoning upon *which computation and which data is the most suitable in any moment when decision is needed*. The right side of the Fig.1 takes the data about and from the computations in the left part and generates ontological concepts for reasoning. This mediation, *judges* how far we go in empowering edges and how fog/cloudlets/clouds help in creating intelligent edge computing. Consequently, we may create a powerful and constantly changeable computing continuum. Figure 1 defines component-based and layered software architecture for mediation and generates software applications with component-based technologies and tools (Juric and Madland, 2020).

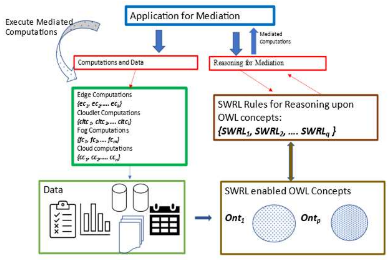

Figure 1. Conceptual Model for Mediation

The semantic of the context where instances of edge computing may exist and be captured, requires either taxonomies or ontological concepts which would represent this semantic and enable reasoning. This is similar to (Juric, 2019) where computational abstraction defines the semantically rich context and its presentation through SWT languages. We could mediate on

(a) Which computations will run on the edge /fog/ clouds and why?
(b) Which data will be used and where do they come from?
(c) Where the results of computing will be held and why?

(a)-(c) is an illustration of questions we may ask our semantic mediator when configuring an intelligent computational edge. If the reasoning upon a particular *context* can answer questions like (a)-(c) above, we can assume that this semantic mediation may serve a variety of contexts and instances of edge computing or the IoT.

3.1. Taxonomy for Creating Reasoning

There are many options for creating taxonomies, but Table 1 shows a particular choice of taxonomical elements we may want to have. The roles of elements stored in the left column of Table are self-explanatory Hardware, Network, Software Architectures, Data, Computations (aligned with software architectures) and Applications (derived from software architectures). The right column in Table 1 gives a set of terms which explain taxonomical elements. These are excerpts from the full-scale listing which illustrates what we may want to have or learn about the *context* in which we wish to mediate. This is by far not an exhaustive list, it can change according to the vision we may have when mediating, but it is a starting point when we try to describe the *context*.

Table 1: Excerpts from the Taxonomy

Hardware	Device, device profiling (sensing, collecting, computing, quality aware, communicating) server, micro-server, fog, cloudlets, edge, no server, routers, switches
Network	Nodes edges, dependencies
Software Architectures (SA)	SA styles, infrastructures, platforms, components, containers, VM, programing framework
Data	Data rates, volume, storage, quality, congestion, Data optimization partitioning, indexing and clustering, Data extracting, cleaning, training testing, Data aggregation, architecting, modelling and engineering
Computations	Types (searching, decision making, reasoning, prediction, learning) Finding, searching for and discovering Algorithms, Computability of algorithms and load balancing.
Applications	Context aware, self tuned, analytics, testing and learning, service delivered, resource effective, QoS, cost of implementation, software overhead,
Sustainability of the Edge	Objectives (accuracy, quality, efficiency effectiveness privacy), Computational efficiency, Resource management, Latency and proximity to data source, Energy consumption, efficiency and sharing, Cashing (popularity and local intelligence), control (blockchain), partitioning and sharing computing tasks.

Note: the taxonomical element *Sustainability of the Edge* in Table 1 stores the data which are more likely to be constraints as opposed to the facts used in reasoning. The constraints conceptualize relationship between taxonomical elements and can be converted into properties required by reasoning languages. Therefore, taxonomical elements from Table 1 can be converted into ontological concepts typical for OWL, but the right column can be used for finding OWL individuals and defining relationships between them.

There is a slight overlapping between the content of the boxes in the right column of Table 1. This division depends on *"what we want from the edge computing in a particular moment"*. This is prone to change and dictated by the *context* within which we wish to conceptualize edge computing. By no means does the right column in Table 1 contain final elements. It is an indication of how to systemize knowledge through the taxonomy. Finally, the structure of Table 1 can change, and its content grow according to operational environments when we configure. edge computing

4. Reasoning for Semantic Mediator

4.1. The OWL Model

The OWL model in Figure 2 is created from the left column of Table 1 with two exceptions. The *Sustainability of the Edge* element is used for defining OWL object properties and thus enabling reasoning. The Edge/F/C element contains all possible configuration elements of instances of edge computing, which may spread to Fogs and Clouds.

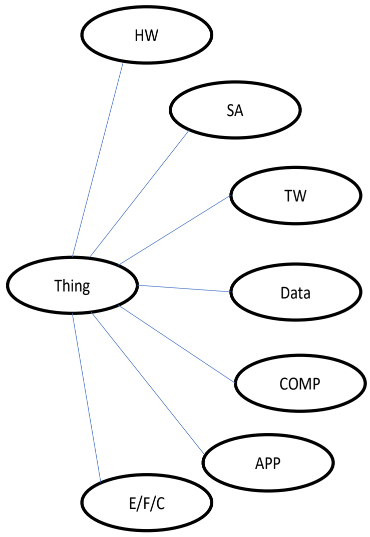

Figure 2 Basic Ontological Model

4.2. The Scenario

The scenario for illustrating the taxonomy, and the proposed OWL model is placed in the healthcare domain. It describes a situation where we try to predict the spread and outcome of an infectious disease (such as covid) during pandemics in a local or reginal environment. The mediator should decide if it is possible to create a computational model at the edge, by running ML clustering and classification algorithms mostly on mobile devices, in proximity of data sources, which supply data for predictions. The mediators should opt for clouds only in circumstances when we need access to global data or look at trends from the environments outside local/regional. The scenario is placed in a Public Health Organization, (PHO) which

a) collates local/regional data,
b) makes local decisions on how to address the spread of infection through prediction,
c) compares results of predictions with actual data, if available,
d) decides which algorithm to run for predictions, when and why and
e) decides which data is relevant for running predictions.

Therefore, a PHO chooses prediction algorithm(s), runs it(them) as frequently as needed, but rely on local/regional data at any moment, while decisions must be made locally and promptly. There is more. The nature of collected data may affect the choice of algorithms. By looking at local figures/ numbers, a PHO may find that they

(i) do not have data set(s) big enough to run predictions and/or
(ii) may have an excessive amount of missing data which does not help in predictions.

There is a correlation between data sets collected and choices of computations a PHO may use upon that data. This is where human intervention may be essential because of constantly changing contexts during pandemics. If we wish to build an efficient mediator for configuring computational edge, under constant changes, humans have a role to play. Could a mediator advise on how to configure edge computing?

4.3. Defining the Context for Mediation

In the numbered steps below, we examine various situations within a PHO, which would require ML predictions for decision making. Each step would specify the context in which computations would happen on the edge, fog or clouds.

1. Continuously receiving data on infection, hospitalization, deaths, locations, and demographic data. Monitoring the situation - **EDGE**
2. Running frequent and ad-hoc predictions with classifiers. Data set is small, predictions not very reliable, they must be repeated - **EDGE**
3. Performing cluster analyses of local data. The size of data set is relatively small / **EDGE**
4. Identifying categories in clustered data, i.e. defining local data categories - **EDGE**
5. Increasing the number of data sources which are not available locally. Adding environmental factors, weather, movement of population, traffic, pollution. Data Collection – moving to **Fog/Clouds** and running classifications on **Fog/Cloud**
6. Monitoring the size of local data – if data increases, investigate potential data cleaning, address missing data and data anomalies – **EDGE**
7. The size of data Rapidly increased – **CLOUD**
8. Writing code for efficient data cleaning for running it on a mobile device - **EDGE**
9. Making decision on priorities: cleaning data versus manual feature selections, versus precisions of classification - **EDGE**
10. Experimenting with impact of clustering on the choice of classifier. Experimenting with clustering and classification – **EDGE**
11. Finding local trends for the epidemic without looking at archived data, i.e. running trends- **EDGE**
12. Finding global trends– **CLOUD**
13. Looking at archived data - **CLOUD**
14. Archiving local data – **Edge, Fog /Cloud**
15. Cutting the size of data set and working with the smaller data size (using human intervention to achieve better precision in prediction, through intelligent feature selections) - **EDGE**
16. Performing data cleaning, but the data set size is excessive – **Edge / Fog / Cloud**
17. Choosing classification algorithm(s) AFTER seeing data – **EDGE**
18. Data dictates the choice of algorithm. Investigation – **EDGE**

We can conclude from the above that we will probably leave the Edge and run ML algorithms on either Fog or Cloud only in special circumstances when

a) the data of interest is not stored locally. We compute close to the data source (acquiring global trends and archives, or processing global data)
b) the size of data is excessive, and we require a high level of precision in prediction. Are we sure that it might be safer to go to clouds
c) Archives have to be created.

In the case of b) we can still perform predictions on mobile devices (EDGE) with the RAM>=16GB, by manipulating feature and data type selections for predictions. This would require human intervention to increase the precision of predictions, even if we reduce the size of data sets. Complex algorithms for data cleaning should be revisited and we should consider a trade-off (algorithm precision, versus feature selection, versus limited data cleaning, versus human intervention). Would the data cleaning really improve the precision and trustworthiness of the results? The mediator should be able to understand situations described in the bullets above (1.-18. and (a)-c)) and make decisions on whether to compute at the edge or somewhere else.

4.4. Creating the Reasoning Process

The reasoning process, as a backbone of the semantic mediation is created through the SWT paradigm and the reasoning upon SWRL enabled OWL ontologies. The SWT dictates the following steps:

a. Converting the semantic stored in Table 1, Figure 1 and the context from 4.3. into OWL classes, individuals an object properties.
b. Choosing OWL classes and object properties relevant for defining a reasoning process.
c. Conceptualizing SWRL rule(s) which would secure reasoning
d. Illustrating the reasoning through an example

There are three important remarks.

First, the data for populating the ontology and defining object properties could be inserted into the mediator automatically from the taxonomy, as experimented in (Shojanoori 2013), (Kataria (2011), but the preference should be to allow a certain user input through an interface. Running predictions and making decisions relevant for a PHO should not be heavily automated. Human intervention reduces the need for running predictions on clouds and increasing trust into algorithms and their results (Juric and Ronchieri, 2022).

Second, the semantic from Table 1 and contexts in section 4.3. are rather complex, and it is not assumed that we must conceptualize it fully in a computational model. The semantic is **to choose from** and changes from one moment to another.

Third, we should also address professional practices of PHOs where flexibility and addressing constant changes in real life situations during any pandemic must be supported by the mediator. Therefore, a PHO should be able to impose on the mediator the exact *context* from one moment to another, in order to address prescribed procedures of PHO decision making processes.

4.5. An Example of a Reasoning Process

There are various possibilities for creating generic reasoning processes in this scenario. Figure 3 shows possibly the simplest model which would be easy to relate to the problem domain. It populates OWL model with the individuals and object properties we can relate to the content of Table 1, Fig. 2 and steps from section 4.3.

The model conceptualizes

- the taxonomical elements from Table 1 into Context Parts (CP) classes $CP_i \in \{CP_1, CP_2, .., CP_n\}$.which can be identical to ontological concepts from Fig. 2.
- the class named ALL CONFIG (in Figure 2 named Edge/Fog/Cloud) to store all possible computational configurations, which in our scenario could be in line with ALL CONF = {Edge, Fog, Cloud, Edge-Fog, Edge-Cloud, Edge-Fog-Cloud, Fog-Cloud}.and
- a set of object properties $OPi \in \{OP_1, OP_2, .., OP_n\}$. between individuals of the ALL CONF class and the set of CP_i classes $\{CP_1, CP_2, .., CP_n\}$.

Object Properties OP_i are denoted as blue bidirectional arrows, and the reasoning is a combination of ontological matching between the individuals of CP_i classes and the individuals of *ALL CONFIG* classes, through *n* different SWRL rules (denoted as orange, blue and green dashed lines in Figure 3). The reasoning would ***infer*** (black line) new individuals in *RESULT* class by moving a set of selected individuals from *ALL CONFIG*, into RESULT class.

According to this model, object properties carry significant semantic, and it would be prudent for the PHO to decide if object properties will be a) asserted manually through user intervention b) inserted automatically from a dedicated persistent data store c) inferred from a different reasoning process. In this model, properties are not inferred, but if we assume that we have constantly changeable contexts, then this would be worthwhile considering.

Note: the choice of object properties is dictated by the problem the PHO must solve through reasoning.

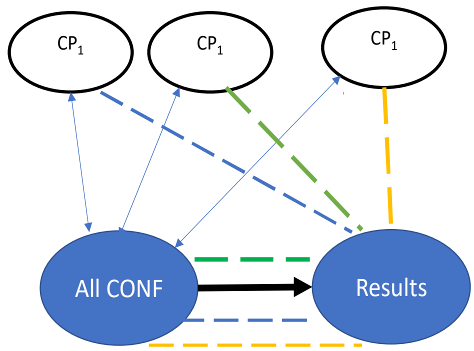

Figure 3 The Reasoning Process

Table 2 gives excerpts from ***OPi*** definitions. The individuals of the OWL classes have been taken from Table 1 and the scenario of the PHO computational needs. For reasons of clarity, we simplified the choices of ***CPi*** into Hardware, Data and Computations and name object properties according to PHO expectations from the computations in one moment (steps from 4.3.

Table 2 shows a few individuals of Domain classes HW, Comp and Data and object Properties OP_i which are defined between **All Conf** individuals and individuals of **HW**, **Comp** and **Data** respectively. Orange arrows are *OPi* named "not-advisable", blue arrows are *OPi* named "is-feasible" and green *OPi* are named "recommended for". It is obvious that these *OPi* are our design decisions dictated by the reasoning process. They will change if for any reason the nature of individuals of all domain classes changes. This design decision is maybe biased towards edge, due to just a few “orange arrows”, but *OPi* are susceptible to context changes, and they do need more attention in conceptualization of the reasoning

Table 2 Excerpts from Object Property Definition

Domain	Indiv.
HW	Phone <16
	Tablet
	Laptop
	sensor
Comp	Read
	Collect
	Cluster
	classify
	Archive
Data	Small
	excessive
	trend

Indiv	Range
Edge	All Conf
Fog	
Fog/ Edge	
Cloud	
Fog/ Cloud	

The reasoning is performed with the SWRL rule below, which moves individual *Edge* from the ***All Conf*** into the ***Results*** class. This will happen in situations where "*we wish to perform reading, collecting and clustering of relatively huge data set which does not have to be archived, but we have a powerful device at the edge (table tor laptop)*.” This result of reasoning is valid in that moment and possibly longer, if there are no changes in this context. Any change in the context invalidate the result of this reasoning (object properties might change!).

HW(?a) & Comp(?b) & Data(?c) & AllComf(?d) &recommended(?a,?d)& is-feasible(?b,?d) &recommended (?c,?d) -> Result (?d))

Figure 4: SWRL Rule for Mediator

5. Related Work

At the time of writing there were no publications which challenge the ideas from Figure 1. and section 4. However, there are a few papers on merging the IoT and intelligent computational edge and address the creation of a computing continuum.

The world of the IoT, which is often associated with the ECP, is becoming more complex and finding solutions which would come close to our research interest is almost impossible. We suggest one paper from CACM (Bouguettaya et al., 2020) which raises the awareness of the scale of complexity of the IoT and uses a traditional view of the IoT, where devices, or things, take center stage. There is one sentence in this paper which does not appear in any other and says that “the management of things” could be facilitated through abstractions defined through IoT services. This is the first step forward towards a computational continuum: through the abstractions of the IoT domain.

The second paper is the analysis of the dependability of edge computing (Bagchi et al., 2020), but this is a rather a depressing read. It analyzes possible malfunctioning of edge computers, decentralized management of devices at the edge and multi-tenancy of resource constrained edge devices. It is no surprise that the heterogeneity and interoperability jumps out as an additional burden to this complicated world of the IoT.

However, there are papers with more optimistic outlooks towards creating a formal ECP (Patel, 2017) (Sheth, 2016), (Greengard, 2020). They are interesting reading but not very close to the ideas from this paper.

When searching for papers on mediation in the IoT, we find attempts to talk about semantic mediation (Chio et al., 2019), (Ali and Chong, 2019), (Dernaika et al., 2020), (Gryk and Ludascher). Their authors are taking the old ideas of mediation, from the mid-90s and use them in 2020 to address semantic heterogeneities, They use mediations for the same purpose as 30 years ago. There are no attempts to create a semantic mediator to generate the best possible configuration of the allocation of computations across edge/fog/cloud for a given situation and move towards a computing continuum.

There are two papers which use the SWT and SWRL to reason upon IoT architectures and context aware IoT applications (Chen et al., 2020) (Maarala, 2016). They do use SWRL and RDF/OWL languages for reasoning, but create formal ontologies which are, by their nature, complex and therefore become a heavy burden for any software application which deploys them. Therefore, the authors’ debate the performance of a *single, distributed, mobile and hybrid reasoner* is immaterial, because it applies to the creation of the semantics stored in formal ontologies. It is the scale of ontological concepts stored in formal ontologies which create a bottleneck in any edge computing. Using ontologies and reasoners upon their concepts, in detecting anomalies in the IoT is another example where heavy apparatus of knowledge, stored in the ontologies, affects the IoT. It is almost impossible to imagine that solutions such as (Chen et al., 2020) would create an instance of IoT which would create intelligent computational edge and drive us towards computing continuum. Formal ontologies as knowledge bases are not the answer for dynamic and constraint edges and thus Figure 1 remains unchallenged.

5.1. Applications of the Proposal

It is important to note that our proposal from Figure 1, which claims to offer layered and component-based software architectures for hosting mediation, does not differ too much from the ideas of breaking a tight integration between software applications, middleware services and operating systems as it was prevalent in mid- 2000 (Mechitov and Agha, 2012). The paper maybe uses old fashioned terminology, appropriate for the time when it was published, but this type of separation of concerns in modern software engineering is still the only way forward. Software components from Figure 1 are the only way of enabling the flexibility of adopting new context *as we go* and mediating about the best possible edge configuration.

Considering that the scenario in section 4.2. is in the domain of running ML algorithms upon alphanumerical data collected locally for a PHO, possibly in proximity of mobile devices, the excepts from the taxonomy in Table 1 do not show data generated through sensory technologies or live and user generated data on social media, which grow extremely fast. These data sets qualify to be *known* to the mediator. In both cases we might need additional elements in the taxonomy (boxes in the right column of Table 1) to enable decision making by a mediator: *Is it prudent to use scalable dynamic programming algorithm*, such as DROPLET from (Elgamal et l., 2018) which partitions computations across shared edge or not? The mediator can decide to avoid partitioning and use reasoning to decide if edges give a sustainable solution considering characteristics of data sources, queuing delays within one and amongst different data streams, heterogeneity and geo distribution of computing resources, tradeoff between communication and computations delays and pipeline parallelism, to mention just a few. The work described in (Elgamal et l., 2018) is an ideal source of data to test the spanning of edge and cloud resources when processing real time streaming data, using semantic mediation from this proposal.

The same applies to the ideas from (Sandur et al., 2022) where query partitioning for near data processing is not recommended, because it is computationally expensive (with query optimization). Therefore, the authors use model-based heuristics to improve the partitioning. The semantic found in server monitoring with adaptive near data processing in (Saundur et al., 2022) is extremely rich and can create another set of inputs in our taxonomy. Examples are: reducing amount of data for stream processing; processing streams on the server nodes; overprovisioning of computing resources: monitoring query to be restricted only on a subset of operations (e.g. filtering), minimizing interference with hosted services, grades of operator level partitioning, changes in node resource conditions and many more. Reasoning upon these taxonomical elements may produce similar results as algorithm JARVIS (Sandur et al., 2022) does, and the mediator may create a stable query partitioning for each *context* it detects when monitoring server workloads. It can change resource allocation when the context changes.

In summary, the semantic mediator from this paper may have applications in enabling computational edge across many domains. By preparing a mediator to give reasoning concepts, their individuals and object properties (i.e. working on a domain specific taxonomy) as specified in section 3, we can use the idea from section 4 and mediate on how to spread computations between edges and clouds in many problem domains.

6. Conclusions

This paper re-iterates the discussion on the problem from pervasive and mobile computing, which heavily depend on cloud computing. The intent is, to move away from clouds and look at all other possibilities of using the space between them and computational edges. We also strive for intelligent edge computing and expect it to accommodate algorithms which shape our current AI. It is unlikely that edge devices will ever replace powerful clouds, but we should use the space between edges and clouds in contexts and mediate which computations would run where and why. This is the first step towards an imaginative computing continuum and mediation through logic and NOT predictive inference. The mediator uses facts and not predictions, i.e. it is not based on predictive inference and would be a light-weight software solution which can give repetitive results at the edge whenever context changes. Running a simple logic inference with SWRL enabled OWL ontology is not a burden for any edge.

This research deliberately leaves physical aspects typical of the IoT, cyber physical and pervasive computing outside the focus. It pushes forward conceptualizations and abstractions which deliver software. If we wished to overcome current computing problems in the world where clouds dominate, we must conceptualize an ECP and not just focus on technology (Nelson and Metaxatos, 2017) (Juric, 2019).

Finally, the proposal should not be confused with formal ontologies, which store and retain knowledge because of reasoning. Our reasoning is a part of a software engineering solution, where a mediator is in its core, programmed through logic reasoning with OWL and SWRL and implemented with Java technologies. Consequently, we have no formal ontologies, we have a sleek and effective computations (short SWRL coding) deployable within component-based software architectures using Java technologies (Tarabi and Juric, 2018). The ontological model is simple and would not need an excessive number of individuals. Results of reasoning is never stored as persistent ontological data because it changes as context changes. It would be very dangerous to keep reasoning results which are not true anymore. The implementation of the proposed model could be done in line with (Patadia et al., 2011), (Shamoug and Juric, 2017), (Juric and Ronchieri, 2022) (Almami et al, 2016) These publications explain the development of software applications from software architectures in line with Fig. 1 and the reasoning similar to Figures 3 and 4.

7. References

Abbas, N. Zhang, Y.. Taherkordi, A. Skeie, T. 2018. Mobile Edge Computing: A Survey. *IEEE Internet of Things Journal* 5(1) pp 450-465.

Ali, S.; Chong, I. (2019) Semantic Mediation Model to Promote Improved Data Sharing Using Representation Learning in Heterogeneous Healthcare Service Environments. *Appl. Sci.*2019, *9*, 4175.

Almami, E., Ahmed, M.Z. Juric, R. (2016) Software Applications Built Upon SWRL Enabled OWL Ontologies, 21st SDPS 2016 Int. Conf. 2016.

Angel, N.A.; Ravindran, D.; Vincent, P.M.D.R.; Srinivasan, K.; Hu, Y.-C. (2022) Recent Advances in Evolving Computing Paradigms: Cloud, Edge, and Fog Technologies. *Sensors* 2022, *22*, 196.

Bagchi, S., Siddiqui, M.B., Wood, P., Zhang,H. (2020) Dependability in Edge Computing, Communications of the ACM, 63(1), pp. 58-66.

Balouek-Thomert, D., Renart, E. G., Zamani, A. R., Simonet, A., & Parashar, M. (2019). Towards a computing continuum: Enabling edge-to-cloud integration for data-driven workflows. *International Journal of High-Performance Computing Applications, 3*(6), pp. 1159 1174.

Bouguettaya,B., Sheng, Q.Z., Benatallah, B., Ghari Neiat A., Mistry, S., Ghose, A., Nepal, S., Yao. L. (2021. An internet of things service roadmap. Communications of. ACM, 64(9). pp. 86–95.

Carvalho, L. I. da Silva, D. M. A. Sofia, R.C. (2020). Leveraging Context-awareness to Better Support the IoT Cloud-Edge Continuum, *2020 Fifth Int. Conf. on Fog and Mobile Edge Computing (FMEC),* pp. 356-359

Chen, G., Jiang, T., Wang, M., Tang, X., Ji,W. (2020) Modelling and reasoning of IoT architecture in semantic ontology dimension, Computer Communications

Chio, A., Bouloukakis, G., Hsu, C. H. Mehrotra, S., Venkatasubramanian, N. (2019) Adaptive Mediation for Data Exchange in IoT Systems, 18th Workshop on Adaptive and Reflexive Middleware, pp 1-6

CISCO What is Edge Computing, CISCO, available at https://www.cisco.com/c/en/us/solutions/enterprise-networks/edge-computing.html

Deng, S. Zhao, H. Fang, J. Yin, J. Dustdar, S. Zomaya Y. (2020). Edge Intelligence: The Confluence of Edge Computing and Artificial Intelligence, *IEEE Internet of Things J.* 7(8) pp. 7457-7469. Volume 153, 2020, pp. 580-594.

Dernaika, F., Cuppens-Boulahia, N., Cuppens, F., Raynaud, O. (2019) Semantic Mediation for A Posteriori Log Analysis. ARES '19, Aug 2019, Canterbury, United Kingdom.

Elgamal, T., Sandur, A., . Nguyen, P., Nahrstedt K., Agha, G. (2018) DROPLET: Distributed Operator Placement for IoT Applications Spanning Edge and Cloud Resources, in Proceedings of the *IEEE 11th International Conference on Cloud Computing (CLOUD)*, 2018, pp. 1-8.

Greengard, S. (2020), AI on Edge, Communications of ACM, 64(9) pp 18-20.

Gryk, M R. Ludäscher. B. (2018). Semantic Mediation to Improve Reproducibility for Biomolecular NMR Analysis. Transform Digit Worlds (2018). 2018 March ; 10766: pp. 620–625.

Hao, Y. Li, M. Wu, D. Chen, M. Hassan, . M. Fortino, G. (2020) Human-Like Hybrid Caching in Software-Defined Edge Cloud. *IEEE Internet of Things Journal,* 7(7).pp. 5806-5815.

Juric, R., Kuljis J., Paul, R. (2004), Software Architecture Style for Interoperable Databases, 26th Int. Conf. on Information Technology Interfaces, ITI 2004

Juric, R., Kuljis J., Paul, R. (2004). Software Architecture to Support Interoperability in Multiple Database Systems. 22nd International Conference on Software Engineering, IASTED 2004.

Juric, R., Ronchieri E. (2022) Semantic Framework for Practicing Data Science in Public Health Organizations during the Covid-19 Pandemics; to appear in the Proceedings of the HICSS 55

Juric, R. (2016). Could Semantic Web Technologies Create New Computational Models Outside Semantic Web, 21st SDPS 2016 Conference.

Juric, R., Kim, I, (2017) Software Architectures for Smart Applications Which Merge Ontological Reasoning with Big Data Analytics, 22nd SDPS 2017 Conference

Juric, R., Ronchieri E. (2022) Semantic Framework for Practicing Data Science in Public Health Organizations during the Covid-19 Pandemics; to appear in the Proceedings of the HICSS 55

Juric, R. Madland, O. (2020) Semantic Framework for Creating an Instance of the IoE in Urban Transport: A Study of Traffic Management with Driverless Vehicles, IEEE ICHMS 2020 Conference, online, Sept. 2020.

Juric, R. (2019). Semantic Model for Creating an Instance of the IoT, IEEE CyberScieTech 2019 Conference,

Kataria, P. (2011). Resolving Semantic Conflicts Through Ontological Layering. PhD Thesis University of Westminster, London, UK.

Lane, N. D. Bhattacharya, S. Mathur, A. Georgiev, P. Forlivesi C. Kawsar, F. (2017). Squeezing Deep Learning into Mobile and Embedded Devices, *IEEE Pervasive Computing*, 16(3) pp. 82-88.

Lopez, P.G. Montresor, A. Epema, D. Datta, A. Higashino, T. Iamnitchi, A. Barcellos, M. Felber, P. Riviere, E. (2015).

Edge Centric Computing: Vision and Challenges, SIGCOMM Comp. Comm. Rev 45(5), pp 37-42
Maarala, A.I., Su, X., Riekki, J.(2017) Semantic Reasoning for Context-Aware Internet of Things Applications, *IEEE Internet of Things Journal*,4(2), pp. 461-473.
Mahdavinejad, M. S. Rezvan, M. Barekatain, P. Adibi, A. Sheth, A. (2018) Machine learning for internet of things data analysis: a survey Digital Communications and Networks, 4(3) pp. 161-175.
Mechitov K. and Agha,G. (2012) Building portable middleware services for heterogeneous cyber-physical systems, *Third International Workshop on Software Engineering for Sensor Network Applications (SESENA)*, 2012, pp. 31-36.
Merenda M. Porcaro, C. Iero, D. (2020). Edge Machine Learning for AI-Enabled IoT Devices: A Review, Sensors 2020, 20, 2533
Milojcic, D. (2020). The Edge-to-Cloud Continuum, *Computer,* 53(11) p.16-25.
Mukherjee, M. Shu, L. Wang, D. (2018). Survey of Fog Computing: Fundamental, Network Applications, and Research Challenges, *IEEE Comm. Surveys & Tutorials,* 20(3) pp. 1826-1857, third quarter 2018
Nelson, S.A. Metaxatos, P. (2016) Design Thinking: The internet of Things Needs Design, Not Just Technology, Harvard Business review, April 29, 2016.
Pirmagomedov, R. Koucheryavy. Y. (2021). IoT Technologies for Augmented Human: a Survey. *Internet Things* 14 (2021): 100120.
Pisani, F., Borin, E. (2018). Fog vs. cloud computing: should I stay or should I go? In: Proceedings of the Workshop on Intelligent Embedded Systems Architectures and Applications, pp. 27–32 Patadia, R.; Kataria, P., Juric, R., Kim, I. (2011) Conceptual Design Of Semantic Software Applications" 16th International Conference on System Design and Processing Science, SDPS 2011.
Patel., P. Ali, M.I., Sheth, A. (2017) On Using the intelligent Edge for IoT Analytics, IEEE Intelligent Systems, Volume 32, pp. 64-69.
Raisamo, R. Rakkolainen, I. Majaranta, P. Salminen, K. Rantala, J. Farooq, A. (2019). Human augmentation: Past, Present and Future, International Journal of Human-Computer Studies, Vol. 131, pp. 131-143
Rausch, T. Dustdar, S. (2019). Edge Intelligence: The Convergence of Humans, Things, and AI, *2019 IEEE International Conference on Cloud Engineering (IC2E)*. pp. 86-96.
Sandur, A. Park, C., Volos, S., Agha G., Jeon, M (2022) Jarvis: Large-scale Server Monitoring with Adaptive Near-data Processing, *IEEE 38th International Conference on Data Engineering (ICDE)*, 2022, pp. 1408-1422.
Saracco, R. (2018). Disruptive Technologies in human augmentation impacting beyond 2040, blog at https://cmte.ieee.org/futuredirections/2018/04/29/disruptive-technologies-in-human-augmentation-impacting-beyond-2040-v/ IEEE Future Directions
Satyanarayanan, M. (2017). The Emergence of Edge Computing, *Computer*, 50(1) pp. 30-39.
Satyanarayanan, M. Davies, N. (2019) Augmenting Cognition Through Edge Computing, *Computer*, 52(7) pp. 37-46.
Satyanarayanan, M. Chen, Z. Ha, K. Hu, W. Richter W. Pillai, P. (2014). Cloudlets: at the leading edge of mobile-cloud convergence, *6th Int. Conf, on. Mobile Comp. Applications and Services,* pp. 1-9
Seal, A. Mukherjee. A. (2018). On the Emerging Coexistence of Edge, Fog and Cloud Comp. Paradigms, in Real-Time Internets-of-Every Things which operate in the Big-Squared Data space. *SoutheastCon 2018*: 1-9.
Sheth, A. (2016). Internet of Things to Smart IoT Through Semantic, Cognitive, and Perceptual Computing, *IEEE Intelligent Systems,* 31(2) pp. 108-112.
Shi, W. Dustdar, D. (2016). The Promise of Edge Computing, *Computer*, 49(5) pp. 78-81.
Shojanoori, R. (2013).Towards formalization of situation-specific computations in pervasive computing environments, PhD thesis University of Westminster, London, UK
Shamoug, A., Juric, R. (2017) Software Tool For Semantic Resources Allocation In Humanitarian Crises, 50th HICSS Conference.
Shojanoori, R., Juric, R. (2014) Ontology Design For Supporting Decision Making In Self Care Homes In Proceedings of the 47th HICSS Conference, HI, US, January 2014
Taleb, T., Ksentini A., Frangoudis, P.A. (2019). Follow-Me Cloud: When Cloud Services Follow Mobile Users, *IEEE Transactions on Cloud Comp,* 7(2) pp. 369-382.
Tarabi M. and Juric, R. (2018) Software Architectures for Smart Applications in The Management of Chronic Diseases: A Case Study of Reversibility Of Diabetes 2, In Proceedings of the 51st HICSS Conference January 2018.
Vaquero, L. M. Rodero-Merino, L. (2014). Finding your way in the fog: Towards a comprehensive definition of fog computing, *ACM SIGCOMM Computer Communication Review,* 44(5) pp. 27-32
Verbelen, T. Simoens, P. De Turck F. Dhoedt,B. (2012). Cloudlets: Bringing the cloud to the mobile user, *Third ACM workshop on Mobile cloud computing and services. ACM,* pp. 29-36
Wiederhold.G. (1995) Mediation in information systems. ACM Computing Surveys, 27(2):265-267.
World Wide Web Consortium: Semantic Web "Layer Cake", https://www.w3.org/2004/Talks/0412-RDF-functions/slide4-0.html
W3C OWL Working Group: OWL 2 Web Ontology Language Document Overview (2012), http://www.w3.org/TR/owl2-overview/
Zhang, B. Mor, N. Kolb, J. Chan, D. S. Goyal, N. Lutz, K. Allman, E, Wawrzynek, J. Lee, E, Kubiatowicz, J. (2015), The Cloud is Not Enough: Saving IoT from the Cloud, The 7th USENIX Conference on Hot Topics in Cloud Computing, pp 21-28.