International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Investigation on Constraints and Recommended Context Aware Elicitation for IoT Runtime Workflow

[1]V. Suresh Kumar, [2]Sanjeev Kulkarni, [3]Naveen Mukkapati [4], Abhinav Singhal, [5]Mohit Tiwari, [6]D. Stalin David

Submitted: 20/08/2023 **Revised**: 08/10/2023 **Accepted**: 21/10/2023

Abstract: Various technological and application challenges arise in the advancement of Internet of Things. Apart from Design and Deployment, Security also falls as a primary challenge to overcome. Device Management is becoming more complex as numerous network services to be handled. Inter-device communication for technical aspects appears to be underappreciated. There is a critical necessity to include this area's requirements and challenges. By abstracting data models and operations and expressing them using semantics, M2Mcommunication and interoperability and may be made simple. A thorough investigation of the foregoing is preparing the way for various approaches. Along with Semantics, a high-level language construct is suggested that can enable run-time workflow construction. Things Markup Language is the name of the concept (TML).

Keywords: *Ontology, Semantic methods, QoS aware, Context aware, Run-time workflow, Interoperability, Things Mark-up Language.*

1. Introduction

The networking model which includes millions of different devices which are very much aware about the surroundings and these networked devices with unique IDs interacting with other machines/objects, with the purpose performing some applications / services is referred as the "Internet of Things." A bio-chip can be provided with Unique ID and the same can be fit with a farm animal. If the bio-chip has the ability to communicate over a network, it becomes an example for Internet of Things. Similarly, a heart monitor implanted inside a human or an automobile with built-in sensors to alert the driver regarding the tyre pressure, or any other natural or man-made object that can be assigned an IP address and given the ability to transfer data over a network are all examples of things in the Internet of Things. A comprehensive understanding of an interactive IoT model is shown in Fig: 1.

Apart from server technology, data management, storage space management, security, and privacy are the major challenges that arise when embedded devices are integrated into the Internet of Things. The design requirements of the new class of embedded devices were incompatible with existing Internet technologies and protocols. Because these embedded devices are often built for cheap cost and low power consumption, they have very limited power, memory, and computing resources, and to save energy, they are routinely turned off for significant periods of time (sleep intervals).

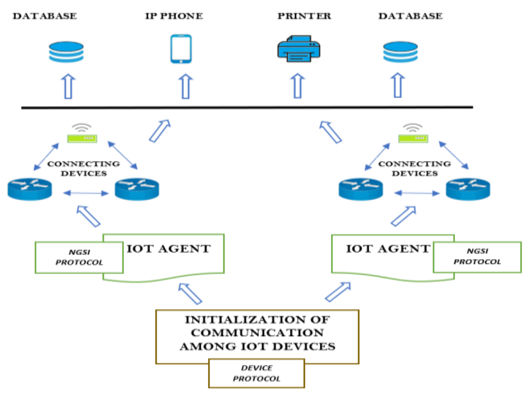

Fig.1: IoT – An interactive model

The expansion of Internet protocol technology to new sectors is speeding up, with embedded devices such as sensing devices playing a significant role. This Internet expansion is equivalent in extent to the Internet's rapid growth in the 1990s, and the resulting Internet is now known as the Internet of Things[1].

1Professor, Department of Information Technology, Vel Tech Multi Tech Engineering College, Chennai, India
akshmikumarvs@yahoo.co.in
2Associate Professor, Department of Information Science & Engineering, Yenepoya Institute of Technolog, Mangaluru, Karnataka
sanjeev.d.kulkarni@gmail.com
3Assistant Professor, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation- 522302
naveenkumar105@gmail.com
4Assistant professor, School of Sciences, Christ (Deemed to be University) Delhi NCR, Ghaziabad-201003, Uttar Pradesh, India.
ism.abhinav@gmail.com
5Assistant Professor, Department of Computer Science and Engineering, Bharati Vidyapeeth's College of Engineering, Delhi
mohit.tiwari@bharatividyapeeth.edu
6Associate Professor, Department of Information Technology, Vel, Tech Multi Tech Engineering College, Chennai, India
sdstalindavid707@gmail.com

In addition, the networks produced by these embedded devices differ from those found on today's Internet. Different traffic patterns, high packet loss, low throughput, frequent topology changes, and tiny viable payload sizes characterize these restricted networks.

The beginning of 21st century saw Kevin Ashton, cofounder and executive director of Massachusetts Institute of Technology's Auto-ID Center, coined the word ‘Internet of Things’ and defined the Internet of Things' latent potential [2]. "Today, computers — and, by extension, the internet — rely on people for practically all of their information. Humans obtained and created virtually all of the roughly 50 petabytes (1,024 terabytes) of data available on the internet by typing, clickinganon/off button, snapping a discrete snapshot, or reading a bar code. User’s time, attention, and accuracy are all restricted, thus they aren't very good at gathering information about the real objects. If the computing systems are powerful enough to collect the data without human interference, everything could be tracked and counted, resulting in huge reductions in waste, loss, and cost. We would be able to tell if something needed to be replaced, repaired, or recalled, as well as whether it was new or old.

The future challenge will be ensuring that the social and technological benefits of the Internet of Things revolution are shared by all people throughout the world, not just those in the world's most developed and expanding economies. The possibility of networking the smart objects paved way for fantasy start-ups like "the Smart Toaster"[3] and "the Smart Coke Machine,"[5].This technical advancement opened the new avenue for smart communication. The IoTsystems / devices were born as a result of this technological achievement.

2. Challenges in IoT

While the above-mentioned potential applications and key concerns are intriguing, the expectations imposed on the underlying technology are significant. In order to equip a large number of devices, a technology must be available, and it must be inexpensive. Scalability, 'arrive and operate,' interoperability, service discovery, software complexity, enormous data and its interpretation, security and personal privacy, fault tolerance, and, most crucially, the power necessary for communication are all difficulties that must be addressed. An abstraction of the IoT difficulties is required and provided. [6].

*Scalability:*This has to do with the ability for a large number of "things" to communicate seamlessly and for service discovery to occur efficiently regardless of the environment.

Arrive and operate: Because the "things" in question aren't computers, there's no need to configure or adapt them to specific scenarios. Instead, relationships must be made on the spur of the moment.

Interoperability: Every smart device in the Internet of Things must contain varied communication capabilities and information processing. Common norms and standards are necessary to enhance communication and cooperation, which is especially crucial when it comes to object addresses. As a result, it becomes unavoidable in terms of adherence to a specified structure. This must be compatible with the existing IP standard.

Discovery: In changing situations, the "things" must automatically identify the appropriate services. It becomes unavoidable that proper semantic techniques of characterising their functionality be developed.

Software complexity: Although smart object software systems will have to operate with limited resources, as in traditional embedded systems, a more extensive software infrastructure on the network and on background servers will be required to manage smart objects and provide services to support them.

Data volumes: Because these sensor devices must operate in real time, they must deal with large amounts of data stored on networked nodes and centralised servers.

Data interpretation: It is critical to interpret the local context supplied by sensors as precisely as possible. Conclusions must be reached quickly from the various data that will be generated.

*Overall Security and individual privacy:*In the Internet of Things, communication confidentiality, authenticity and trustworthiness of communication partners, and message integrity are all critical.

Fault tolerance: The IoT's dynamic state necessitates a rapid shift in context in unforeseen ways. The user wants to be able to rely on things working properly. As a result, the Internet of Things (IoT) must be constructed in a reliable and trustworthy manner, with redundancy at several levels to ensure adaptability.

Power supply: In the Internet of Things, smart gadgets must be fuelled by a self-sufficient energy source. Because of their size and weight, batteries and power packs are issues in many gadgets. They also lose energy as a result of their maintenance requirements. RFID transponders are smaller, less expensive, and do not require their own power supply.

Short-range and wireless communication: Wireless communication across a few centimetres will suffice in the Internet of Things. Because of the short-range connection, just a little amount of electricity is needed. Induction coupling is used in NFC and RFID.

3. Interoperability in IoT

To achieve uniform programmability of devices or sensors, the IoT ecosystem requires interoperability. This stresses the importance of IoT standards in enabling horizontal platforms that are communicative, operable, and programmable across devices of any make, model, manufacturer, or industry. The objective is to connect people, processes, and things without regard for screen type, browser, or hardware.However, in reality, the Internet of Things is fragmented and suffers from a lack of interoperability; smart devices with diverse or overlapping solutions are difficult to link. As a result, it has an impact on how the gadgets interact.

Interoperability

"The ability due to networking of smart systems or devices or sensors with varied attributes to interchange data and utilise information" is referred as Interoperability. This concept poses numerous obstacles in terms of obtaining information, exchanging data, and utilising information in order to comprehend and interpret it[4]. Interoperability across several layers, such as ‘device connection’ to ‘seamless integration’ to ‘data provided by IoT resources’, is a problem for the global expansion of generic IoT systems.

H. van der Veer et al., discussed about the ability of various categories of interoperability (due to heterogeneous hardware devices, various software systems, and different platforms)to enable machine-to-machine communication[5]. This falls under the heading of 'Technical Interoperability,' which is primarily concerned with (communication) protocols and the infrastructure required to support them. The term'syntactical interoperability' is typically connected with data formats. High-level syntaxes like HTML or XML can be used to represent the data formats. The term "semantic interoperability" refers to persons sharing a common understanding of the meaning of the material (information) being transferred. 'Organizational Interoperability,' as the name suggests, is the ability of organisations to effectively communicate and transfer meaningful data (information) despite the fact that they may be using a variety of different information systems across a variety of infrastructures, and probably across diverse geographic regions and cultures. Successful technological, syntactical, and semantic interoperability are prerequisites for organisational interoperability.

The primary problems [7] for interoperability, and our focus, are (i) various data source integration, (ii) Peer-to-Peer (P2P) Communication and (iii) Semantic Interoperability (unique ontological point of reference). The first identifies the need for data/event interoperability so that data/events from various data sources can be combined/aggregated more readily. With respect to ontology, Semantic interoperability refers a single point of reference. Hiring a third party to translate across several methodologies or merging/mapping ontology can help solve this problem. There may also be protocols for agreeing on a specific ontology. The third point is that apps must be able to exchange specialised information at a higher level. At lower levels, interoperability may be disregarded, but it can be applied at a higher level.

Data Modelling has three models: (i) Conceptual data model, (ii) Logical data modeland (iii) Physical data model. Various Data Exchange formats are there. Few of them are (i) XML, GML (ii) CSV (iii) JSON (iv) Apache Parquet (v) YAML (vi) REBOL. Different ways of Knowledge Representation are (i) Logical Representation (ii) Semantic Network Representation (iii) Frame Representation and (iv) Production Rules. Likewise Data/Event Semantic Annotation and Ontology merging / matching & alignment have various tools for merging and alignment. As disclosed above, along with various data models, different data exchange formats and different ways of representing the knowledge bring in more anomalous semantic challenges that require much more research. In addition to the three issues mentioned earlier, the above mentioned are other significant challenges while dealing semantic interoperability.

Review on Semantic Interoperability Challenges

IoT demands the support of a wide range of devices, protocols, and procedures because it incorporates both old and new technology. In order to address the interoperability issues, the focus should be on the development of protocols, designs and frameworks, promotion of standards,and media-type standards, and leveraging abstract interface defining languages and semantic technologies. Because it is impossible to produce a standard or specification that can meet all needs, none of these approaches has been successful in resolving the interoperability problem.

In order to manage numerous volumes of IoT/M2M devices, various open source languages, reference architectures, standards and protocols were developed by the consortium of Vendors.

IPSO defines a RESTful framework[9] for addressing CoAP/HTTP sensors/devices. Function Sets, which are well-defined URIs, are used to address sensors. Different root-paths were configured and released to access the networked smart things/Sensors/Devices, GPInput / Output, energy required for functioning, Memory, Load scheduling, and Balancing, Light Control, Message, Location, and Configuration. For example, the command '/dev', displaysthe metadata corresponding to the smart device, which includes the detailed information about the manufacturer, model, and serial number of the smart device.The IETF CoRE Group Communication set the definition for the various types of resource, data, and interfaces. The end point will be set using the configured function-sets which in turn, gain knowledge of about configuration options, and keep track of sensor occurrences.

The ETSI/oneM2M project group proposed a standard for end-to-end M2M communications as deliberated by Shelby Sensinode et al., [10]. The design of interoperable utlizationis done with end to end requirement with the help of functional architecture. Service capabilities, related reference points, functional entities as well as information model, security, management, billing and implementation guidelines. OneM2M also has a public ontology that may be used to interface with other systems. Semantics can automate the discovery, interpretation, and application of M2M data from a variety of sources, allowing higher-level services to be created.

Table 1(a): Literature Survey I

ARTICLE	FINDINGS	SHORTCOMINGS
A Context-aware Computing Mediated Dynamic Service Composition and Reconfiguration for Ubiquitous Environment[21]	1. Abstract BPEL is defined for dynamic service composition and reconfiguration. 2. Context awareness decision making is used to select services based on ontology	1. BPM is required to provide abstraction to cater to many diverse types of devices, services 2. Abstract BPEL defined works for SOA with SOAP bindings, but most of the IoT frameworks are RESTful. QoS includes other parameters than just user preferences
Research on IOT RESTful Web Service Asynchronous Composition Based on BPEL [22]	RESTful IoT web-service architecture for Asynchronous execution support is added into BPEL	Async process invocation is only discussed, handling the response and composing the flows is not discussed
Modeling BPEL-based Collaborations with Heterogeneous IoT Devices[23]	Adaptors for different device types in BPEL	Adaptors for BPEL is by nature still very static, it cannot be used to compose dynamically
Internet of Things-aware Process Modeling: Integrating IoT Devices as Business Process Resources [24]	Using native software components as resources in workflows and processes	BPEL is by nature still very static, it cannot be used to compose dynamically BPEL with native service integration may have limited application as there is diverse set of device often in a given workflow

SWE is a decoupled method that will enhance its capacity to address IoT needs with a variety of decision workflows [11].SWE is again a collection of Web Service interfaces that abstracts the complexities of the sensor network connection. The term 'Sensor Web' is also used to describe a sensing system that largely relies on the Internet.

IoT-A/OpenIoT deals device management with no explicit support for workflow[12]. IoT – A forum proposed an Architecture Reference Model (ARM) which opened the way for developing a standard platform for IoT applications.IoT ARM should establish a uniform framework and set of rules for dealing with the essential features of IoT system development, use, and analysis.

IoT-A allows developers to make architectural decisions that fit the device they're working on while also providing them with recommendations to follow. The IoT ARM is a mechanism for establishing IoT system interoperability, not a guarantee of interoperability between any two specific system designs. OpenIoT is a middleware that forgets information from sensor clusters without knowing the sensor type. It is based on the IoT-A ARM.

AllJoyn/IoTvity - Custom data-models[13]: AllJoyn is an open-source software platform for programmes that can locate and communicate with neighbourhood devices without the need of the cloud, regardless of brand, category, method of transportation, or operating system.It is designed to work on a wide range of platforms,including small embedded RTOS platforms and full-featured operating systems. There are varieties

Table 1(b): Literature Survey II

ARTICLE	FINDINGS	SHORTCOMINGS
Web-of-Objects Based User-Centric Semantic Service Composition Methodology in the Internet of Things[26]	1. Object virtualization and semantic ontology based service composition 2. New composite service can be formed, searched and re-used in an efficient manner using natural language queries	1. Service composition cannot be made only using Semantic reasoning, other factors like Context, QoS need to be considered 2. Service virtualization is required since different there are a number of different service interfaces
Dynamic Workflow Composition using Markov Decision Processes [27]	1. Markov Decision Process to model workflows which accounts for the environment and dynamism 2. Stochastic and Bayesian model for learning the model and workflow building mechanism	IoT has other criteria like QoS, context awareness that need to be considered in the decision process and Bayesian model
QoS aware web service composition using genetic algorithms[28]	1. Qos based web-service selection using genetic algorithm 2. Qos based resource allocation and scheduling for multi cloud env.	IoT may not too many choices for service selection to start with, however ontologies can be used to improve accuracy in IoT
Design of a Situation Aware Service for Internet of Things[25]	1. Provides a Self Organizing software platform for fault recovery and detection	Self organizing workflows are not discussed, they just discuss about specific service recovery

of language bindings and transports available (BLE,Zigbee, Z-Wave). One of the benefits of the AllJoyn design is that smart things can be found on the local network,without having to connect to the internet, minimising the number of devices that need to be linked to the internet.

IoTivity is a competing framework that enables data discovery, transmission, and management and device management services, comparable to AllJoyn. Data transmissionallows for information sharing and control.It is based on a messaging and streaming architectural style and Data management entertains the aggregation, storage, security and assessment of data from many different sources. Configuration, deployment, and diagnostics are all possible with device management.

LwM2M mainly considers the factors such as low bandwidth and lossy networks[14]. Open Mobile Alliance LwM2M specification promotes the object model for interoperability.While dealing with the smart Internet of Things (IoT) devices, issues like limited bandwidth and lossy networks need to be considered. These issues can be dealt through Lightweight M2M (LwM2M) which is a device management protocol. The interface between M2M devices and servers are taken care by LwM2M application protocol which uses CoAP/Datagram Transport Layer Security on UDP and Short Message Service for the purpose. REST-based device management architecture, and client registration functions, service enablement and reporting, and asynchronous notification controlare defined by LwM2M.

SenML is a data-model in JSON format[15]. Without some metadata, sensor data in its raw form cannot be comprehended in an exact manner.SenML does not require any additional metadata or schema in order to create generalized apps meant for a variety of purposes.SenML uses the JSON format, which provides a nice balance of use and efficiency. SenML is a recognized IETF standard which provides excellent result for designing efficient, simple, autonomous interoperable applications.

	Pros	Cons
BPM/ BPEL	1. Provides the necessary abstraction, methods and processes model 2. Synchronous process/service execution support	1. BPEL doesn't provide mechanism to use metadata – so it lacks context awareness data (time, location) 2. BPEL doesn't provide a way to replace one service with another at runtime (static binding and definition) 3. BPEL doesn't provide support for asynchronous execution

Table 2: Pros and Cons of BPM/BPEL

Weave's objective is to produce distributed deliberations across robots, people, and smart devices. [16].Weave is an IoT device common language that allows gadgets, humans, and smartphones to communicate in a single language. It is based on pre-defined standards for defining device characteristics. Weave was designed from the roots up to be cross-platform, allowing a smartphone to control a household appliance that is running on a different platform.Weave supports all kinds of application stacks as well as Android stacks.

Thread - secure and reliable connection[17]: Thread is a mesh network-based protocol for connecting hundreds of goods throughout the home in a secure and dependable manner. Thread is primarily intended for usage in Wifi mesh networks in houses.The core of interoperability is 6LoWPAN, which is generally supported protocols and IPv6 technology.It's developed to work with a wide range of household appliances, apart from access control, climate control, energy management, lighting, security, and safety.Thread is another interoperability-focused protocol designed by a conglomeration of companies but deployment will be difficult due to the hefty software and hardware changes required.

Vorto is a tool for defining all of the devices' capabilities [18, 19]. The information model is a feature in Vorto that allows us to characterize all of the device's capabilities. This data model is kept in a common repository so that suppliers can utilize it to create solutions.As a result, the M2M interoperability choices are highly fragmented. Because of this and various different devices are involved, developing a process is challenging.

Runtime workflow require-ments/ solution	IPSO	ETSI / oneM2M	SWE	IoT-A Open IoT	ALLJoyn / IoTvity	LwM2M	SenML	Weave	Thread	Vorto	Ponte
Protocol	✘	✘	✘	✘	✘	✔	✘	✔	✔	✘	✔
Semantics	✘	✔	✔	✘	✘	✘	✘	✘	✘	✘	✘
Message	✘	✔	✔	✘	✘	✘	✘	✔	✔	✘	✘
Properties	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✘
Behavior	✔	✔	✔	✔	✔	✔	✘	✔	✔	✔	✘

Table 3: Run Time Workflow composition in IoT

In Table 3, IoT Workflow Composition during Runtime with respect to Protocols, Standards, Semantics, behaviour, reference designs, and interface definition languages have significantly abstracted away the many differences. While developing applications and solutions for various services, the interoperability has to be provided more importance. This will enhance the advancement of Internet of Things to a greater extent. This, in turn, leads to define a proper designat the application layer for constructing a High Level Language. This construct can well be utilized to automatically bring in the dependency at various layers of the application stack, allowing standard interfaces to be exposed that are aware of communication protocols, data kinds, and other considerations. Such tactics can promote autonomous system development while simultaneously assuring interoperability.

4. TML – A New Approach towards Semantic ontology integration

The above interoperability concerns can be resolved by using Things Markup Language to abstract the many differences in data representations and service interfaces. Communication between two or more entities of a system could be achieved by enumerating the numerous unlikeliness in interfaces in service, data type, and communication between two or more entities of a system using a English like Language Construct called as Things Markup Language.

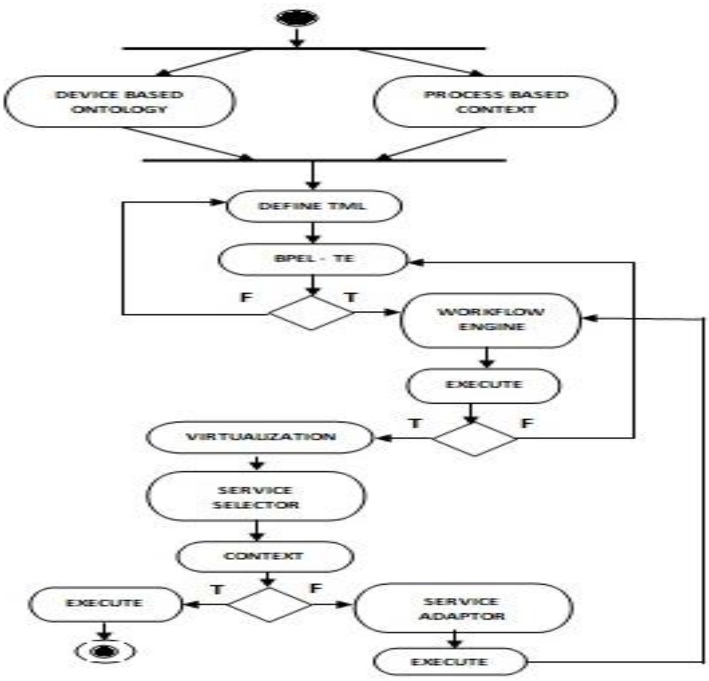

Fig 2: Context aware workflow Execution – Activity Diagram

The following suggestions are made to address the aforementioned concerns to some extent, if not entirely, and it is believed that they will surely open up a plethora of new and imaginative avenues for delivering effective solutions.

Context Aware programmes that adapt their behaviour according to the user’s current context – where they are, who they are with, what time of day it is and so on. To be more explicit, "Contextual Awareness" refers to the utilisation of user-specific data, such as Location-based Technology and sensors, by a computing device to determine its user's conditions. Currently, the majority of such applications are created by experts in research labs. Context-aware systems are concerned with acquiring context, abstracting and comprehending context and implementing application behaviour depending on the recognized context.

Interoperability is essential when various systems/devices require to communicate with each other due to differences in discovery, service interfaces, capabilities, data models, and methodologies. The basic data models include physical data models, logical data models, and conceptual data models, each with its own set of criteria. Identifying interfaces aids in the definition of system boundaries. Cost overruns and product failures are frequently caused by missing or improperly designed interfaces. The ability to identify interfaces ensures that the system and any other systems that need to interact are compatible.

Ontology represents Internet of Things resources, entities and services. The usage of a unique ontology allows IoT systems to be represented and used without requiring unnecessary processing time while querying. Ontology is a domain-specific framework for describing shared and reusable information. They are the foundation for modelling high quality, linked, and coherent data because of their capacity to describe relationships and high interconnectivity. The Semantic Ontology interconnection will provide context awareness and "what" the data signifies.

TML is a high-level programming language construct that covers the various disparities between data models and service interfaces. The Things Markup Language (TML) allows run time workflows to use context and semantic knowledge to determine the optimal services for a given time, event, and location - workflow models have to be aware of the ontology (operating environment).The detailed flow of work is depicted in Fig 2 via an activity diagram.

TML can be used to establish a run time workflow irrespective of the type or make of the underlying implementation device. Straight forward and clear remarks can be used to recognize, locate, and contact a service interface, as well as to ascertain context, transition between data types/models, analyse the data, and draw conclusions regarding data values and context. Selecting the proper device for a particular circumstance and implementing the desired service on it is referred to as matchmaking.

Although many devices have limited resources, compiling run time procedures is necessary for native service. To maximise the service selection from a list of prospective services, ontology with a hybrid model incorporating Bayesian learning must be implemented.Response-Time,Cost, User Preferences Throughput, Reliability, Reputation, and Availability, must all be considered when executing this service. Run time (reactive or dynamic) workflow building, reconfiguration, and execution are essential when objects/services must be updated during runtime.

5. Conclusion

Many standards were produced after the real solution was developed, and in many situations, the standard is not broad enough to be used. When designing apps and solutions for the Internet of Things, interoperability should be prioritized by developers at all application stacktiers. At the application layer, a generalised specification for a high-level language construct can be defined. This high-level language construct can be used to create bindings automatically at different levels of the application stack, allowing standard interfaces to be exposed that are aware of communication protocols, data types, and other aspects. Such techniques can both encourage autonomous system development and provide compatibility.

References

[1] IsamIshaq et al., “IETF Standardization in the Field of the Internet of Things (IoT): A Survey”, Journal of Sensor and Actuator Networks, ISSN 2224-2708, Vol 2, p 235-287; 2013.

[2] Kevin Ashton, Auto-ID Center, MIT, The Internet of Things - presentation to Procter & Gamble, 1999.

[3] "The Internet Toaster." Living Internet, 7 Jan. 2000.http://www.livinginternet.com/i/ia_myths_toast.html. Accessed August 2016

[4] "The "Only" Coke Machine on the Internet". CarnegieMellonUniversity.https://www.cs.cmu.edu/~coke/history_long.txt Accessed on August 2016.

[5] Tschofenig, J. Arkko, D. Thaler, D. McPherson, RFC 7452, “Architectural Considerations in Smart Object Networking”, ISSN: 2070-1721, 2015.https://datatracker.ietf.org/doc/html/rfc7452

[6] F. Mattern and C. Floerkemeier. From the internet of computers to the internet of things. Volume 6462 LNCS, pages 242 – 259, 2010.https://vs.inf.ethz.ch/publ/papers/Internet-of-things.pdf

[7] IERC - European Research Cluster on the Internet of Things: IoT Semantic Interoperability: Research Challenges, Best Practices, Recommendations and Next Steps, March, 2015.http://www.internet-of-things-research.eu/pdf/IE

[8] RC_Position_Paper_IoT_Semantic_Interoperability_Final.pdf

[9] H. van der Veer, A. Wiles, "Achieving Technical Interoperability – the ETSI Approach", ETSI White Paper No.3, 3rd edition, April 2008. https://www.etsi.org/images/files/ETSIWhitePapers/IOP%20whitepaper%20Edition%203%20final.pdf

[10] Z. Shelby Sensinode, C. ChauvenetWatteco, "The IPSO Application Framework", IPSO Alliance Interop Committee, 24 August 2012.https://www.omaspecworks.org/wp-content/uploads/2018/03/draft-ipso-app-framework-04.pdf

[11] "OneM2M architecture", Technical Specification, Doc. No: TS -0001-V1.6.1, 30 January, 2015. https://onem2m.org/images/files/deliverables/TS-0001-Functional_Architecture-V1_6_1.pdf

[12] Mike Botts, 'SWE-IoT Workshop, Open Geospatial Consortium", Texas, USA, 21 March 2012.

[13] https://portal.ogc.org › files/?artifact_id=52162

[14] "IoT-A ARM", SOTA report on existing integration frameworks/architectures for WSN, RFID and other emerging IoT related Technologies, 4 April 2011, Accessed Aug 2016

[15] David, D. S., Anam, M., Kaliappan, C., Arun, S., Sharma, D. K. et al. (2022). "Cloud Security Service for Identifying Unauthorized User Behaviour", CMC-Computers, Materials & Continua, 70(2), 2581–2600.

[16] "Light Weight M2M", Open Mobile Alliance - Enabler Test Specification for Lightweight M2M Candidate Version 1.0 – 29 August 2016. https://www.openmobilealliance.org/release/ETS/OMA-ETS-LightweightM2M-V1_0-20160829-C.pdf

[17] Xiang Su, Hao Zhang, JukkaRiekki, Ari Keränen, Jukka K. Nurminen, Libin Du, "Connecting IoT Sensors to Knowledge-based Systems by Transforming SenML to RDF, Procedia Computer Science, Volume 32, 2014, Pages 215-222, ISSN 1877-0509.https://www.sciencedirect.com/science/article/pii/S1877050914006176?via%3Dihub

[18] "Weave", https://developers.google.com/weave, Accessed Aug 2016

[19] "Thread Stack Fundamentals", Thread Group – Public Release, Revision 2, July 2015.https://www.silabs.com/documents/public/white-papers/Thread-Stack-Fundamentals.pdf

[20] "Vorto", http://www.eclipse.org/vorto/Accessed Aug 2016

[21] "Vorto Repository", http://vorto.eclipse.org/,Accessed Aug 2016

[22] "Ponte", http://www.eclipse.org/ponte/,Accessed Aug 2016

[23] Tao Gong, Zheng Hu, HaiFeng Liu, Feng Lin, Dian Zhou and Hui Tian, "A context-aware computing mediated dynamic service composition and reconfiguration for ubiquitous environment," *2012 3rd IEEE International Conference on the Internet of Things*, Wuxi, 2012, pp. 16-23. doi: 10.1109/IOT.2012.6402299

[24] L. Zhang, S. Yu, X. Ding and X. Wang, "Research on IOT RESTful Web Service Asynchronous Composition Based on BPEL," *2014 Sixth International Conference on Intelligent Human-Machine Systems and Cybernetics*, Hangzhou, 2014, pp.62-65.DoI: 10.1109/IHMSC.2014.23

[25] S. D. Kim, J. Y. Lee, D. Y. Kim, C. W. Park and H. J. La, "Modeling BPEL-Based Collaborations with Heterogeneous IoT Devices," *2014 IEEE 12th International Conference on Dependable, Autonomic and Secure Computing*, Dalian, 2014, pp. 289-294.DoI: 10.1109/DASC.2014.58

[26] Sonja Meyer, Andreas Ruppen, and Carsten Magerkurth. 2013. Internet of things-aware process modeling: integrating IoT devices as business process resources. In *Proceedings of the 25th international conference on Advanced Information Systems Engineering* (CAiSE'13), Camille Salinesi, Moira C. Norrie, and Óscar Pastor (Eds.). Springer-Verlag, Berlin, Heidelberg, 84-98. DOI: http://dx.doi.org/10.1007/978-3-642-38709-8_6

[27] Im Y. Jung and et al., "Design of a Situation Aware Service for Internet of Things", International Journal of Distributed Sensor Networks, Volume 2015, Article ID 641312, 8 pages. http://dx.doi.org/10.1155/2015/641312

[28] Safina Showkat Ara and et al., "Web-of-Objects Based User-Centric Semantic Service Composition Methodology in the Internet of Things", International Journal of Distributed Sensor Networks, Volume 2014, Article ID 482873, 11 pages. http://dx.doi.org/10.1155/2014/482873

[29] Prashant Doshi, "Dynamic Workflow Composition using Markov Decision Processes", Source: DBLP, January 2005, DOI:10.1109/ICWS.2004.51

[30] Mahmood Allameh Amiri and Hadi Serajzadeh, "QoS Aware Web Service Composition Based on Genetic Algorithm", 5th International Symposium on Telecommunications, 2010, pages 502 -507.

[31] Mwangi , J., Cohen, D., Silva, C., Min-ji, K., & Suzuki, H. Improving Fraud Detection in Financial Transactions with Machine Learning. Kuwait Journal of Machine Learning, 1(4). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/148

[32] Ibrahim, S. S. ., & Ravi, G. . (2023). Deep learning based Brain Tumour Classification based on Recursive Sigmoid Neural Network based on Multi-Scale Neural Segmentation. International Journal on Recent and Innovation Trends in Computing and Communication, 11(2s), 77–86. https://doi.org/10.17762/ijritcc.v11i2s.6031

[33] Kshirsagar, P.R., Reddy, D.H., Dhingra, M., Dhabliya, D., Gupta, A. A Review on Comparative study of 4G, 5G and 6G Networks (2022) Proceedings of 5th International Conference on Contemporary Computing and Informatics, IC3I 2022, pp. 1830-1833.