A survey on IoT technologies and their applicability

Emanuel Achirei
Faculty of Computer Science, Alexandru Ioan Cuza University
emanuel.achirei@gmail.com

***Abstract*—Society is undergoing a new industrial revolution which marks the inception of the industry of things. Due to its highly changing nature, we require to take a step back from time to time to analyze the current environment. The objective of this study is the evaluation of the status quo with the goal of gaining insight in regard to developing standards, cloud providers, the impact of this emerging industry and its effects on other domains. This article considers prominent solutions affecting various spheres such as economy, medicine, agriculture etc., how they influence said areas and technical aspects of the product. The results present an incipient maturation of the technological approach, while the adoption factor has a slower ascension due to infrastructure costs and inertia of the targeted industries. This survey highlights the complexity of an IoT environment and compares the features implemented by platforms pertaining to multiple categories. The analyzed solutions provide a rich and distinctive feature set, but it seems there is no absolute winner.**

***Index Terms*—IJOA, Journal, Optimization**

I. Introduction

Starting with monitoring a Coke vending machine at Carnegie Mellon University in 1990, as mentioned in [1], Internet of Things (IoT) has come a long way in the past decades. With an evergrowing number of connected devices the IoT market also requires adapting to its consumers. Hence, many IoT solutions are surfacing every day with the purpose of enriching the context of our decisions. From society, to industry and home use, every domain greatly benefits from IoT integrations.

Nonetheless, all of these solutions require their own infrastructure which provides device management, deployment administration, monitoring etc. While there are multiple kinds of IoT platforms, this report will focus on cloud platforms used as IoT platforms, industrial internet of things (IIoT [2]) platforms as alternatives and open-source options.

The remainder of this paper is organized as follows. The next chapter presents the motivation for choosing the analyzed platforms. Chapters III, IV and V contain the review of the cloud centric solutions, the other trending iot platforms and, respectively, the open-source solutions. The feature summary of the IoT platforms is reported in chapter VI. Conclusions of the paper and future-work directions are provided in chapter VII.

II. Trending solutions for IoT deployment

Due to the widespread usage and the familiarity of the developers, cloud platforms represent a great contender for the list of analyzed solutions in this paper. Since the cloud environment has reached maturity, it makes sense to use the developed standards and apply them to the emergent IoT market. Taking a look at some key players of the cloud environment, Amazon, Microsoft, Google, IBM, their focus on this niche, becoming segment in the last years, is obvious.

Nonetheless, the previous list is built on the assumption that the IoT environment is related to the cloud one. But this hypothesis could be wrong since the number of devices, and the sheer amount of data, might require a whole new paradigm. Ideas that support this theory range from novel technical approaches, such as edge [3] and fog [4] computing, to new business concepts, such as a marketplace for IoT data [5]. Of course, this new perception also raises problems along the likes of data governance, data sovereignty etc. However, this discussion is out of the scope of this paper and was meant to exemplify the different use-cases IoT might impose.

Considering the previous arguments, this survey also includes IoT platforms developed from the ground up, as their approach might be more suitable for the integration of the features mentioned above. The issue with selecting such platforms is the fact that some of them are built as in-house products, not available to the public use.

On the other side of the fence, there is the open-source community, which is just as excited about the new nature of the Internet. Hence, various task forces [6], consortiums [7], [8] and even government funded projects [9] are also working on developing protocols, standards and platforms.

All in all, this paper will analyze products pertaining to all these categories, from the stable cloud developers, to the proprietary solutions developed alongside IoT products and last, but not least, the efforts of the open-source community.

III. Cloud centric IoT platforms

AWS offers 10 services in their Internet of Things category and as an IoT platform its domain seems to be a general one: it offers both services to satisfy a home user, an IoT startup or an industrial site. Their supported protocols range from LoRaWAN [10], HTTPS, to MQTT and MQTT over WebSocket Secure (WSS), encompassing a wide range of use-cases. Besides the security offered by these communication protocols, they also provide a special service tasked with overseeing the best security practices in regard to IoT devices. Another important feature is modelling Things: for an easier process of integrating the same kind of device from different vendors, it could be modelled in a generic way that expresses the basic functionality. With AWS's IoT Things Graph [11], users can develop their own models or use existing ones. Furthermore, this facilitates the creation of flows between different models of Things. Hardware support is provided via FreeRTOS [12], an operating system for microcontrollers that facilitates IoT development, but also through a custom tailored portal for monitoring industrial equipment. Deployment management on the edge is achievable by using AWS's IoT Greengrass [13] through their proprietary agent. Aside agents, integration can also be done through various SDKs ranging from C++, Python, JavaScript, Java to Embedded C and Mobile SDKs. Event processing can be tackled through numerous Amazon services, but another aspect is detecting the state of Things through those events. This can be achieved by either creating detection models for them, or even using machine learning algorithms to further enhance the responsiveness of the solution. Users can access monitoring and analytics data through a diverse set of services and informative dashboards. The access to all these resources can be controlled through the use of IAM roles allowing for fine-grained regulations. Last, but not least, to accelerate development, AWS also offers a marketplace for IoT solutions. All in all, AWS's IoT platform is very versatile, fitting a wide-range of use-cases, while also offering implied scalability through their experience as a cloud platform.

Microsoft Azure offers 24 services in their Internet of Things category, but many of them aren't specific to IoT. Nonetheless, their presence in said list is an indication of the flexibility of the platform offered by Microsoft. Projects developed here can also scale from tens of devices to millions by using some of the well-established protocols such as MQTT, AMQP and HTTPS. They also grant WebSocket support for the first two which ensures higher resilience in regards to transmitting information. The security offered by the Microsoft platform starts from silicone: they provision certified microcontrollers which embed accredited chips safe-guarded by a custom tailored operating system. And in the case of edge computing, they also provide a specialized service which facilitates the management of any computational processes. Building on top of that, they administer authentication for IoT devices and a security overview of the connected equipments. To properly model the functionality of a Thing, a user can use the Azure Digital Twins [14] service which allows for the creation of a digital model of the physical appliance. Deployment management can be easily performed due to the vast range of supported operating systems, list which ranges from Linux to Azure RTOS [15] - Microsoft's custom RTOS. The event management process isn't handled by a specialized IoT software, which hurts the user experience, yet it's still a very scalable and durable data processing functionality. Azure facilitates integration through a plethora of SDKs, covering the most popular programming environments: .NET, C, Java, Node.js, Python and iOS. All these capabilities are easily accessed through their IoT Hub, offering a bird's eye view of the ecosystem, and if a detailed view is required, the monitoring system satisfies any debugging needs. This central point also manages the authorization functionality, through IAM roles which can be manually assigned to their resources. The marketplace is a great place to search for certified devices and, if you are looking for pre-built solutions, Microsoft offers IoT accelerators, which should suite the needs of any application or at least provide a base implementation. In conclusion, Microsoft Azure offers a comprehensive toolset for any IoT project, but their functionalities should be better emphasized for a richer user experience.

GCP's offer of IoT services is more succinct, providing only the Google Cloud IoT API [16], which registers and manages IoT devices that connect to the Google Cloud Platform. They offer basic protocols for communication such as MQTT and HTTP which are relayed through their respective bridges. Security is enforced through the use of JSON Web Tokens signed with the certificates used for provisioning the devices. Thing description is achieved by adding metadata to a device, but no other enrichment is possible. While explicit hardware support isn't specified, GCP provides their own application-specific integrated circuit (ASIC) called Edge TPU [17] which incorporates AI capabilities. Deployment management appears to be a very raw feature, through pushing device configuration via the hub. To process the events sent by the devices, one can use any of the managed services offered by the GCP platform for data processing. With regard to monitoring, a crude overview of the appliances' activity can be analyzed, while further analytics should be configured by the user through the usage of other GCP services. Authorization can be configured at service level, but more fine grained capabilities aren't available. The marketplace offers a handful of solutions when querying for "iot" and no accelerators or pre-built solutions could be found. Google's IoT platform relies heavily on the usage of pre-existing services, while their user experience seems to provide just the building blocks required for designing an IoT environment.

IBM also offers their own IoT platform through IBM Watson IoT [18]. Unfortunately for the purpose of this survey, the platform couldn't be accessed, so the following analysis is based solely on their presentational documents. Their communication protocols list is pretty basic, including MQTT and Java Message Service (JMS) [19], but they offer extensions through custom protocol plugins. Data access and storage is secured through the use of certificates, API keys and authentication tokens. Device management has a base implementation for modelling, but no further details regarding grouping Things, defining them or reusing Thing definitions could be found. Sensors can be updated through commands, but there doesn't seem to exist a custom tailored solution for deployment management. Event streams can be processed through other IBM cloud services, allowing the use of machine learning algorithms for a more thorough analysis. Regarding developer integration, their SDKs are built for C, Java, Node.js and Python. Monitoring and analytics are offered either through custom made dashboards or informational reports. Authorization can be handled by using either pre-defined roles or custom made ones. There doesn't seem to be a solutions marketplace, but their website presents a wide array of plublicly recognized solutions built using IBM Watson. Unfortunately, due to the fact that the platform wasn't accessible for this analysis, we cannot describe a comprehensive outlook of its features. Nonetheless, the documentation presents a well-rounded IoT solution, which accompanied by the various AI services offered can maximize the potential of the data collected.

IV. Other trending IoT platforms

Researching platforms focused solely on IoT can prove difficult. The first reason would be the popularity of the cloud centric platforms and secondly the level of trust: the field of IoT hasn't matured enough to prove the need of an entirely separate kind of platform. However, looking back at its roots of monitoring machines, it makes perfect sense to analyze the IoT platforms used in factories, supply chains etc. Following that train of thought, this paper analyzes the industrial IoT (IIoT) platforms provided by PTC ThingWorx, Siemens MindSphere, Bosch IoT Suite and some interesting alternatives.

At the time of writing this article, PTC ThingWorx offers 5 services, but the greatest difference is the fact that this platform is provided as on-premise: the consumer would install these applications on his hardware and has to manage their availability. Since their domain is focused on IIoT, their protocols range from HTTPS, SMS, SMTP, POP3 to a plethora of protocols integrated as extensions available from their marketplace, and also standard industrial protocols. Their solution embeds security procedures for device management, and a very rich device modelling functionality: a user can define Thing templates, shapes and groups, to ensure the proper modelling of the running environment. Hardware support is heavily emphasized by supporting over 150 industrial protocol drivers which outlines the consideration put in the industrial side of IoT. Deployment management seems to have basic support only via configuration updates and event management is crudely implemented through an alerts system. The SDK area encompasses the following programming languages: C, Java, .NET. While this range is a bit more limited, the Java SDK supports Android development and further interactions can be achieved through the use of the APIs. A monitoring dashboard is available, which presents both a history of device events and also any available logs. These can be further analyzed via statistics, for a better comprehension of the current state. Access control is managed through permissions, which offer fine-grained control over the access a user can possess, while any outliers are present in the audit log. The PTC marketplace administers both extensions to the provided applications and also pre-built solutions ready to bootstrap any IIoT project. Finally, while this IoT platform provides a very rich user experience for an educated user, its features can be a bit overwhelming. Furthermore, the fact that it's presented as an on-premise solution can be a drawback for a consumer which has no pre-existing hardware available, but this can be overcome through the use of cloud platforms. These cloud platforms can be further utilized with the deployed solution through various integrators which facilitate reaching the full potential of the gathered data.

Siemens' platform, MindSphere, is more industrial centric, but they still support popular communication protocols such as HTTPS and MQTT, besides the wide range of lower level protocols. Their security methods aren't explicitly stated, but they claim to support industry standards and device authentication is facilitated through unique identification numbers and security tokens. Their devices can be modelled through either vendor specific designs or generic representations of Things; the vendor specific designs are further supported through their integrations with various microcontrollers and devices. Deployment management doesn't seem to be available in the trial version surveyed here. Users can handle device events by creating flows through an user interface, which contributes to the user experience. Other event processing capabilities don't seem to be available for this review. Integration with the platform is facilitated through Java, C++ and Web SDKs. The devices can be monitored and analyzed either through the MindSphere Fleet Manager [20] or through the Visual Analyzer [21], services which facilitate the use of the data exported by the devices. User and role management are only available for the applications developed on the platform; the environment allows to configure external applications integration with the data available. Siemens also presents a solution marketplace which contains both proprietary solutions and also custom made ones. Siemens' MindSphere packs a rich feature-set, but the user-experience has its shortcomings and without it a potential customer could miss on their extensive functionalities. Yet this downside is alleviated by their extensive range of supported industrial equipment.

Bosch offers its IoT services both as suites and also for standalone use. Besides, the classic communication protocols MQTT, HTTP and AMQP, they also support protocols specifically aimed at IoT such as LoraWAN and CoAP [22]. Security is implemented through the use of certificate authentication and authorization on a per-device basis. The platform relies on open source projects for some of its functionalities, such as Thing management; devices can be described through various models, ranging from generic functionalitites to vendor specific, all available through the Vorto Repository [23]. Hardware support is granted by their IoT Edge [24] services which serves as both a gateway for non-IP IoT devices and also as a computational hub, allowing for event management through flows, analytics and AI enrichment. Furthermore, consumers can remotely command firmware updates and update their deployments. Their SDKs range is narrow but carefully honed: they provide Java plugins for the Eclipse IDE, which expedites the development process. Monitoring is displayed at various service levels, but if more advanced analytics are required, Bosch has developed a custom tailored managed service with the sole purpose of getting the most out of the data collected by the devices. User and role management don't seem to be available for the trial provided, but the option to select and group devices seems to indicate to its existence. Overall, Bosch's IoT platform is very well developed, with an extensive feature-set, refined through a blend of proprietary and open-source projects. Furthermore, there are available integrations with both AWS and Azure. The controversial point is their user experience. On one hand, they offer their services as suites: one looks to be honed for a smaller number of devices, while the other seems to be targeted to an industrial-level of number of devices. On the other hand, the service dashboards are disjointed, and at a first glance it can be tedious to jump from one application to the other.

Cisco, the networking infrastructure leader, is also establishing a foothold in the IoT market. Due to their background as providers of out of the box solutions, this intent is also observed in the matter at hand; their services rely on proprietary software and hardware which allows access to more advanced functionalities, but has the potential to lead to vendor lock-in. Their solution for IoT is called IOx and is described as an application environment that encompasses specific devices and gateways, secured communication via a proprietary operating system and further integrated services. They seem to be less descriptive about specific communication protocols supported, but they mention managing the data through standard protocols such as RPC and PubSub. Through the provided demo, the IoT Operations Dashboard [25] can be accessed, which boasts multiple essential features. Here a user can have a bird's eye view of their devices' location and alerts, while also being able to add new devices. A new device can either use an existing, or a newly created Asset Type [26], which embeds both metadata and also groups multiple sensors in said type. This facilitates the management of the existing devices, but the downside is represented by the sensor catalog, which, at least in the demo version, only offers a handful of Cisco certified sensors. Worth mentioning is their, now at end-of-life, Cisco Kinetic [27] platform, described as an IoT data fabric, which utilized Edge & Fog Processing Modules [28] and also Data Control Modules [29]. Cisco's proficiency is highlighted again by their approach to managing fleets of devices: Cisco IoT Field Network Director [30]. It is described as a complete networking solution capable of monitoring, managing and deploying millions of devices. Developers can integrate with Cisco's IoT services by building IOx Applications [31], described in YAML [32] format and packed as Docker images. Their examples are based on Python and support for other programming languages isn't specified. The authorization angle seems to be tackled through the IOx environment, but specifics regarding granularity aren't obvious. They don't seem to provide a marketplace for prebuilt solutions or accelerators, but product descriptions hint at integrations with various cloud platforms. As mentioned at the start of its description, Cisco doesn't offer a platform, it offers an environment. While their solutions apparently require a hard commitment on the part of the consumer, their time proven experience certainly stands the ground for their IoT services.

Orange, the telecommunication corporation, also launched its own IoT platform called Live Objects. They both provide Internet protocols for communication such as MQTT and HTTPS, while also leveraging their infrastructure for CoAP, LoraWAN and SMS. Besides the inherent security of the communication protocols, they also implement API keys and certificates for authentication. Device management is provided through basic metadata of the device, but it also uses the connectivity capabilities of the appliance to further control the actions the platform can apply. Hardware supported is presented through the implementation of specific communication protocols. Depending on the connectivity of the device, various functionalities are offered: command, configuration or resource. The Things send their data through streams, which further allow for the enrichment or analysis of the events. The state of the devices can be monitored in customizable dashboards and in case of failures, various alerting rules can be implemented. Authorization is implemented through the use of roles backed by pre-defined labels which control the resources to be accessed. Unfortunately there are no marketplace solutions or other extensions to expedite the development process. Overall, the Live Objects platform may seem simple, but with simplicity comes ease of use, and their features have the potential to support the quick development of an IoT solution.

V. Open source IoT platforms

While the industrial giants provide extensive IoT solutions, the open-source community also jumped on the track of this new industry. Ranging from IoT platforms for home use, to production ready IoT platforms, their capabilities shouldn't be overlooked. For the purpose of this paper, the most prominent projects were selected based on their GitHub activity which was used as an indicator of the platform's success.

ThingsBoard is an open-source IoT platform with an extensive feature set. Their supported protocols range from the standard HTTPS and MQTT to OPC-UA [33]. Device authentication can be achieved through access tokens, basic credentials and certificates. Things can be described through custom labels, can have device profiles to automate various actions and they can also be grouped to enrich the user experience. Specific hardware support doesn't seem to be available, besides the plethora of integrations with industrial protocols. Deployment and over the air upgrades aren't supported out of the box, but there are plugins that satisfy this requirement. Rule Engine [34] is ThingsBoard's framework for event processing. Using visual flows, a consumer can greatly enhance the interactivity of the data produced by the devices. No explicit SDKs are offered, as they claim that the simplicity of the API doesn't require it. Nonetheless, a list is available containing community client-side libraries for MQTT and C implementations for CoAP. To monitor the connected devices, a user can utilize the telemetry feature and expose said data using custom made widgets or dashboards. The authorization functionality doesn't seem to be very complex, but solves the access problem in a simple way: tenant users can fully-manage devices and they can create dashboards for customers to visualize their data. There isn't a ThingsBoard marketplace per-se, but the solution offers integration with many cloud providers and also has the contribution of the open-source community. The billing plan is split between the Community Edition which is free and the Professional Edition which boasts some extra features which elevate the user experience. In conclusion, ThingsBoard is a viable option as an IoT Platform, being capable to support the needs of both home users and also enterprises.

OpenRemote is another open-source IoT platform. While it has a lower number of contributors, the overall features presented can satisfy the requirements of an IoT project, and the technologies it's based on are in trend with the current software environment, which reflects the effort put in keeping this project up to standards. The list of supported communication protocols contains the standard HTTPS and MQTT and also future or custom integrations. The security of the system is backed by Keycloak [35], but the process of device provisioning only offers username/password authentication. However, devices can be modelled through an extensive list of provided Assets [36]. Specific hardware support and deployment management is left up to the user, which can implement these functionalities through custom agents. There are no SDKs available, but the main programming language is Java, so a developer could build his own tools if required. Monitoring of devices is offered through the use of a geo-located map which highlights the state of the managed Things. Event management is handled through three options: When-Then [37], Groovy [38] or Flow [39] rules. These allow establishing interactions between the devices present based on a wide array of conditions. The analytics feature is satisfied through the use of custom dashboards, which can be used to analyze the historical usage or state of the assets. Authorization and access management have a basic implementation, allowing the creation of read/write users, but the underlying security system provides the potential for future improvements. OpenRemote has a few gaps to close to achieve production ready state, but its current set of features satisfies the requirements of a home IoT solution.

VI. Feature summary of IoT platforms

This survey analyzes a wide range of IoT platforms and takes note of the commonly required features. A wide array of communication protocols and/or a variety of SDKs allows for a quicker integration process. Every integrated device needs to be authenticated, through a process named provisioning. This functionality was heavily researched as referenced in [40], hence it becomes a requirement for any enterprise level platform. Once the data flow is established, a user could declare himself satisfied with the status quo. However, as the devices grow in number, the need to give meaning to the data arises. An ideal IoT platform should be able to describe a Thing through its functionality; this way, the event processing system can take specific actions and further enhance the interaction between devices and user. Once the system has grown, the user should still be able to notice any problems in the environment. He should be able to monitor and analyze statistics regarding his devices' usage and in case of issues, he should be able to update any of the malfunctioning Things. Of course, from the perspective of the platform, all these actions must be audited and controlled through an extensive authorization system.

All the IoT platforms present in this paper were compared regarding the features described above, and the outcome of this comparison is presented in Table I.

TABLE I
IoT platforms comparison

Feature	**Cloud centric**				**Alternatives**					**Open-source**	
	AWS	***Azure***	***GCP***	***IBM***	***ThingWorx***	***MindSphere***	***Bosch***	***Cisco***	***Live Objects***	***ThingsBoard***	***OpenRemote***
Extensive communication protocols	✓	✓	×	×	✓	✓	✓	✓	✓	✓	✓
Secure provisioning	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×
Thing modelling	✓	✓	×	✓	✓	✓	✓	×	×	✓	✓
Hardware oriented features	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	×
Deployment management	✓	✓	×	×	×	×	✓	✓	×	×	×
Event management	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓
SDKs	✓	✓	✓	✓	✓	✓	×	×	×	×	×
Monitoring	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Analytics	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Authorization mechanisms	✓	✓	✓	✓	✓	✓	×	✓	✓	×	×
Accelerators	✓	✓	×	×	×	✓	✓	×	×	×	×

VII. Conclusions

Since the IoT industry is growing at an accelerated rate, both existing enterprises and start-ups try to capitalize on its potential, which always leads to the question: "Where to deploy the IoT solution?".

Due to the sheer number of IoT platforms (300 in 2018 as mentioned in [41]), only a slice of them were analyzed. This survey was conducted by accessing either trial versions of the platforms, where possible, and/or through the analysis of their publicly available descriptions. Analyzing platforms from different subcategories, a set of common features emerged, which were used for the comparison of said products. While some of these characteristics are more important than others, such as security, some of them cannot be easily gauged. The survey at hand follows a broad use-case of utilization for these environments, so changing this angle of perspective might put the result of this paper in a different light. For example, a consumer interested in implementing an IoT solution for farming equipment won't be affected by the fact that a platform doesn't support the HTTP protocol, as long as there exists hardware support for his devices.

Hence, the answer to the initial question cannot be easily obtained. While some of the cloud centric platforms seem to cover all the features evaluated, their user experience could prove too complicated to handle. On the other hand, if the solution in question is more focused on IIoT, maybe the platforms providing extensive hardware support might fit better; or, if the scope of the task at hand is smaller, such as a home project, an open-source option might be favored.

As such, for future work, these platforms could be reviewed through the perspective of representative use-cases: an enterprise solution for smart-homes, a solution for healthcare equipment or the management of a large-scale farm.

References

[1] "The little-known story of the first IoT device - Industrious." [Online]. Available: https://www.ibm.com/blogs/industries/little-known-story-first-iot-device/
[2] "What is IIoT? Industrial Internet of Things Explained." [Online]. Available: https://internetofthingsagenda.techtarget.com/definition/Industrial-Internet-of-Things-IIoT
[3] W. Z. Khan, E. Ahmed, S. Hakak, I. Yaqoob, and A. Ahmed, "Edge computing: A survey," *Future Generation Computer Systems*, vol. 97, pp. 219–235, aug 2019.
[4] R. Mahmud, R. Kotagiri, and R. Buyya, "Fog Computing: A taxonomy, survey and future directions," in *Internet of Things*. Springer International Publishing, 2018, vol. 0, no. 9789811058608, pp. 103–130. [Online]. Available: https://link.springer.com/chapter/10.1007/978-981-10-5861-5_5
[5] K. Mišura and M. Žagar, "Data marketplace for Internet of Things," in *Proceedings of 2016 International Conference on Smart Systems and Technologies, SST 2016*. Institute of Electrical and Electronics Engineers Inc., dec 2016, pp. 255–260.
[6] "GlobalPlatform IoTopia - GlobalPlatform." [Online]. Available: https://globalplatform.org/iotopia/
[7] IOTC, "Internet of Things Consortium," 2014. [Online]. Available: http://iofthings.org/#members
[8] IIConsortium, "Industrial Internet Consortium," pp. 2017–2019, 2017. [Online]. Available: http://www.iiconsortium.org/
[9] "H2020 — Programme — H2020 — CORDIS — European Commission." [Online]. Available: https://cordis.europa.eu/programme/id/H2020
[10] "Low-Power Wide Area Network (LPWAN) Overview," Tech. Rep., may 2018. [Online]. Available: https://www.rfc-editor.org/info/rfc8376
[11] Amazon, "AWS IoT Things Graph," 2019. [Online]. Available: https://aws.amazon.com/iot-things-graph/?nc2=h_ql_prod_it_tg%0Ahttps://aws.amazon.com/iot-things-graph/
[12] R. T. E. Ltd., "FreeRTOS - Market leading RTOS (Real Time Operating System) for embedded systems with Internet of Things extensions," 2018. [Online]. Available: https://www.freertos.org/
[13] "AWS IoT Greengrass - Amazon Web Services." [Online]. Available: https://aws.amazon.com/greengrass/
[14] "What is Azure Digital Twins? - Azure Digital Twins — Microsoft Docs." [Online]. Available: https://docs.microsoft.com/en-us/azure/digital-twins/overview
[15] "Real Time Operating System (RTOS) — Microsoft Azure." [Online]. Available: https://azure.microsoft.com/en-us/services/rtos/
[16] Google, "Google Cloud IoT Core documentation," 2020. [Online]. Available: https://cloud.google.com/iot/docs/concepts%0Ahttps://cloud.google.com/iot/docs/
[17] GoogleCloud, "Edge TPU - Run Inference at the Edge — Edge TPU — Google Cloud," 2019. [Online]. Available: https://cloud.google.com/edge-tpu/
[18] IBM Cloud, "About Watson IoT Platform," 2018. [Online]. Available: https://console.bluemix.net/docs/services/IoT/iotplatform_overview.html#about_iotplatform
[19] Sun Microsystems, "Java Message Service (JMS)," 2002. [Online]. Available: http://java.sun.com/products/jms/
[20] "Fleet Manager." [Online]. Available: https://documentation.mindsphere.io/resources/html/fleet-manager/en-US/index.html
[21] "Visual Analyzer ™ 4," 2004. [Online]. Available: https://documentation.mindsphere.io/resources/html/visual-analyzer/en-US/index.html
[22] C. Bormann, S. Lemay, H. Tschofenig, K. Hartke, and B. Silverajan, "CoAP (Constrained Application Protocol) over TCP, TLS, and WebSockets," Tech. Rep., feb 2018. [Online]. Available: https://www.rfc-editor.org/info/rfc8323
[23] "Eclipse Vorto." [Online]. Available: https://www.eclipse.org/vorto/
[24] "Bosch IoT Edge: Empower the Intelligent Edge - Bosch IoT Suite." [Online]. Available: https://developer.bosch-iot-suite.com/service/edge/
[25] "Cisco IoT Operations Dashboard - Cisco." [Online]. Available: https://www.cisco.com/c/en/us/products/cloud-systems-management/iot-operations-dashboard/index.html
[26] "Asset Types - Cisco IoT Operations Dashboard - Document - Cisco DevNet." [Online]. Available: https://developer.cisco.com/docs/iotod/#!asset-types
[27] Cisco Inc, "Cisco Kinetic IoT Platform - Cisco IoT Solutions - Cisco." [Online]. Available: https://www.cisco.com/c/en/us/solutions/internet-of-things/iot-kinetic.html
[28] "Accelerate critical decisions and optimize network use with distributed computing," Tech. Rep., 2017.
[29] "Quickly and easily extract data from your devices," Tech. Rep., 2017. [Online]. Available: www.ciscokinetic.com
[30] T. Cisco, "Cisco IoT Field Network Director," pp. 1–4. [Online]. Available: https://www.cisco.com/c/en/us/products/cloud-systems-management/iot-field-network-director/index.html
[31] "Create IOx Application using Docker image to host on access points - Cisco DevNet." [Online]. Available: https://developer.cisco.com/docs/app-hosting-ap/#!develop-and-test-iox-applications/develop-and-test-iox-applications
[32] "YAML Ain't Markup Language (YAML™) 1.0." [Online]. Available: https://yaml.org/spec/history/2003-09-01.html
[33] OPC Foundation, "Unified Architecture - OPC Foundation," 2019. [Online]. Available: https://opcfoundation.org/about/opc-technologies/opc-ua/
[34] "Rule Engine Overview — ThingsBoard Community Edition." [Online]. Available: https://thingsboard.io/docs/user-guide/rule-engine-2-0/overview/
[35] "Keycloak." [Online]. Available: https://www.keycloak.org/
[36] "User Guide: Assets, Agents and Attributes · openremote/openremote Wiki." [Online]. Available: https://github.com/openremote/openremote/wiki/User-Guide%3A-Assets%2C-Agents-and-Attributes#assets
[37] "User Guide: Use When Then Rules · openremote/openremote Wiki." [Online]. Available: https://github.com/openremote/openremote/wiki/User-Guide%3A-Use-When-Then-Rules
[38] A. G. Project, "The Apache Groovy programming language." [Online]. Available: https://www.groovy-lang.org/%0Ahttps://groovy-lang.org/
[39] "User Guide: Use Flow Rules · openremote/openremote Wiki." [Online]. Available: https://github.com/openremote/openremote/wiki/User-Guide%3A-Use-Flow-Rules
[40] S. Nastic, S. Sehic, D. H. Le, H. L. Truong, and S. Dustdar, "Provisioning software-defined IoT cloud systems," in *Proceedings - 2014 International Conference on Future Internet of Things and Cloud, FiCloud 2014*. Institute of Electrical and Electronics Engineers Inc., dec 2014, pp. 288–295.
[41] H. Hejazi, H. Rajab, T. Cinkler, and L. Lengyel, "Survey of platforms for massive IoT," in *2018 IEEE International Conference on Future IoT Technologies, Future IoT 2018*, vol. 2018-Janua. Institute of Electrical and Electronics Engineers Inc., mar 2018, pp. 1–8.