

On the Complexity of Cloud and IoT Integration: Architectures, Challenges and Solution Approaches

Damian Kutzias¹^a, Jürgen Falkner²^b and Holger Kett²^c

¹University of Stuttgart IAT, Institute of Human Factors and Technology Management, Germany

²Fraunhofer IAO, Fraunhofer Institute for Industrial Engineering IAO, Germany

Keywords: Integration, Architecture, Cloud Computing, Internet of Things, IoT, Cloud Architecture, IoT Architecture.

Abstract: Cloud Computing and the Internet of Things (IoT) shift from trend technologies to well established and broadly accepted means to foster business development and service quality. For utilising their full potential in the context of complex systems, applications based on these technologies often have to be properly integrated resulting in major challenges. In this paper, we provide means for better understanding possible occurrences of integration challenges when establishing Cloud and IoT systems. We briefly present several existing Cloud and IoT architectures and a survey on existing integration challenges. Based on these results, we derived an overall integration architecture as a supporting tool for the indication of the different integration challenges, which is presented in a short and full version due to the overall complexity. At last, some general approaches for integration are discussed.

1 INTRODUCTION

Cloud Computing and the IoT are two trend technology paradigms of the last years and also two main enablers for smart manufacturing and digitisation. Smart Products can have advantages over their standard variants and customers begin to expect certain levels of monitoring, smartness and scalability making these technologies more and more a necessity for many enterprises. For production systems, the adoption and integration might be necessary to fulfil the market demands and being able to compete (Chrysolouris et al., 2012).


As usual, with new technologies new challenges arise. The integration of Cloud and IoT into existing hard- and software systems comes with such challenges. Not only solving integration problems, but also knowing about possible occurrences as well as calculating the costs and necessary effort to solve them are challenging tasks.


Cloud-based integration is a system integration that is delivered as a Cloud process and therefore as a service. It includes data integration often implemented with a service-oriented architecture (SOA)


(Korpela et al., 2016). When considering IoT systems, an integrated service can be defined as a service that works with entities (physical devices) and composes them with non IoT services (Thoma et al., 2012).

While traditionally enterprises have bilateral integration of services from point to point, the trend is to chose modular solutions such as an integration Platform as a Service (iPaaS), often making extensive use of API management with micro service architectures. This can solve integration challenges at a fraction of the costs and significantly less time (Pathak and Khandelwal, 2017). Anyway, most enterprises have got legacy systems in use, which usually requires difficult and costly integration or replacing activities (Gholami et al., 2017). For small and medium-sized enterprises in particular, the benefits from using integration of different Software as a Service (SaaS) based on an iPaaS solution are high, also when considering hybrid architectures including existing on-premises services (Bolloju and Murugesan, 2012).

Considering both, Cloud Computing and IoT, the combined usage brings the most benefits due to the storage and computational resources needed for IoT. This makes communication easy and affordable and enables new capabilities (Botta et al., 2014) and therefore lately, these two technologies are converging (Cavalcante et al., 2016).

^a <https://orcid.org/0000-0002-9114-3132>

^b <https://orcid.org/0000-0003-2059-825X>

^c <https://orcid.org/0000-0002-2361-9733>

It is not always useful to store and compute everything in the Cloud. With IoT, the two paradigms Fog Computing and Edge Computing emerge. While the first is about using and optimising the components and resources between the Cloud and the devices, the latter is about using the devices themselves, e.g. for preprocessing the data (Linthicum, 2018). The use of Fog and Edge Computing is considered to be appropriate for many IoT services, e.g. connected vehicles, smart grids and smart cities, and is envisioned to be a unifying platform fostering IoT applications (Bonomi et al., 2012).

As (nearly) always in this context, security and privacy play an important role; due to high sensitivity of data in Cloud and IoT systems, it is critical to develop techniques that enable data integration and sharing without any privacy breaches (Madaan et al., 2018). For instance, a collection of timestamps may not be seen as personally identifiable information or personal data, but it could contribute to data linkage and thereby may have some influence on privacy risks (Danezis et al., 2014).

2 INTEGRATION ARCHITECTURE

Within this section, an overview over several existing Cloud and IoT architectures and their core structure and scope is given. After that, we propose our integration architecture. It is an extensive architecture focused on components and connections relevant for integration challenges. It is differentiated and derived from the outlined architectures as well as possible IoT integration challenges we collected and identified.

A generally accepted basic IoT architecture splits the components of an IoT system into three layers, namely the Perception Layer containing the sensor devices, the Network Layer for the connectivity and the Application Layer for everything related to Services (Wu et al., 2010)(Lee and Lee, 2018). This might be supplemented for example by a Business Layer and a Processing Layer (Wu et al., 2010). From a pure Machine to Machine (M2M) perspective, the European Telecommunications Standards Institute (ETSI) gives a high level architecture consisting of only two layers called Network Domain and Device and Gateway Domain (European Telecommunications Standards Institute, 2013). Mapped to the basic architecture from above, this matches the Network Layer and the Perception Layer. While the Network Domain contains the fundamental requirements, such as network access and the core network, the Device and Gateway Domain also contains special net-

work elements which are M2M or IoT related.

(Douzis et al., 2018) provide a sensor data collection service architecture, which includes several different storages, namely user information, sensor information and a logs for sensors and subscriptions.

There are also existing layer models with an integration focus, especially when an Enterprise or Manufacturing Service Bus or an iPaaS is included for all or major parts of the integration challenges. Schel et al. propose a five layer architecture: 1) Data Source, 2) Data Service, 3) Integration, 4) Integration Service and 5) Business Process with a manufacturing service bus covering the Integration Layer and additional integration services and manufacturing applications on the Integration Service Layer. Whereas from a data centre perspective this might cover all of or the major integration challenges, we have a more holistic view of integration challenges without a focus on smart manufacturing, also including the physical environment and the basic infrastructure as well as message broker systems.

In (Srdjan Krco and Carrez, 2014), a architectural reference model functional view is given as a layered model of functional groups management, service organisation, virtual entity, IoT service, security and communication between the device and the application, giving an overview of the main functions usually required by IoT systems.

For our integration architecture we decided to separate it into three areas and six layers, as shown in Figure 1. The areas are ranging from the shop floor (industrial internet of things) or field (internet of things) through the actual IT system (Cloud or on premises) to the user (that may be anywhere) or actuator (included in machines on the shop floor or in products out in the field). Concerning the six layers you find sensors, users or actuators on the Edge of any Cloud or IoT architecture. Human users or products and machines in the role of actuators make up the top layer. Sensor devices, which might include some kind of Edge processing capabilities, make up the bottom layer. Each of the two requires an Access Layer towards the actual IT infrastructure. The latter is divided into a Data Layer where the actual data processing takes place and an Application / Services Layer where the added value services are located that are to be made use of by the top layer. Figure 1 also shows certain functions for each layer like e.g. Fog pre-processing of sensor information. Finally security integration and IT governance are tasks that span through all layers and therefore need to be addressed holistically.

For integration tasks the Access Layers towards the before mentioned are most relevant, together with

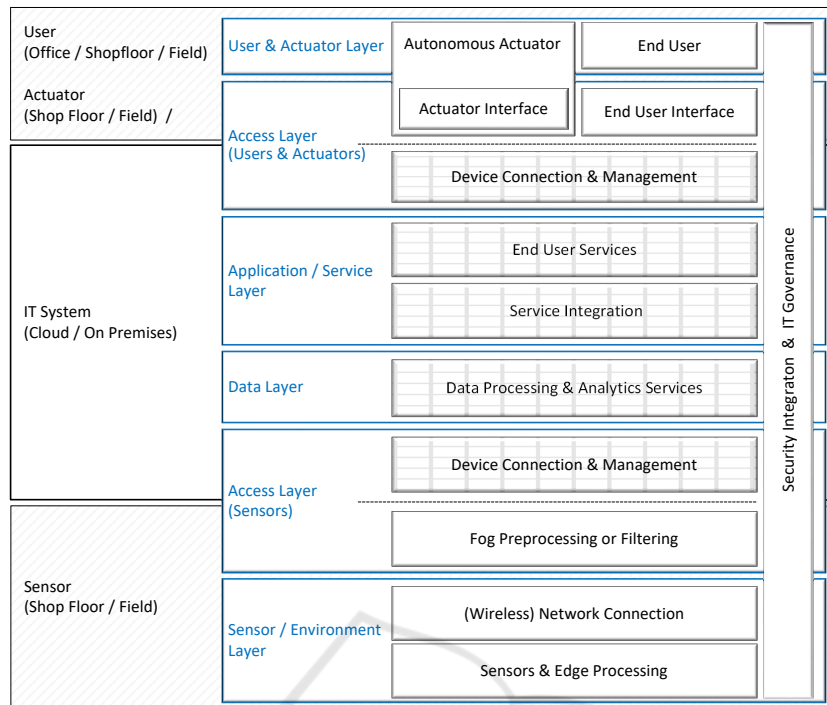


Figure 1: Architecture Overview.

the data Processing Layer and the Application or Service layer. In the latter the data previously collected, stored and managed are refined in order to produce the required results for users and actuators. The information flow goes from bottom to top if you look at Figure 1. This structure matches the enabling technology categorisation to object, networking, middleware and application in (Čolaković and Hadžialić, 2018). Compared to the first mentioned and most fundamental three layer basic IoT architecture, this corresponds with the same basic architecture and an additional Data Layer, which was added. The reason for that is, that there are several data integration challenges and data do not necessarily belong to specific applications for example when it is collected without a use case in mind. Data is often cleaned (Salem and Abdo, 2016) before being used by software solutions. The detailed graphical overview can be seen in Figure 2 and its components and layers are described in the following.

2.1 Our Architecture

A clear and widely accepted definition of IoT platforms or reference architectures is missing and unambiguity usually can only be achieved by abstraction (Guth et al., 2016). However, to be able to provide means to gain a good overview and understanding of complex Cloud and IoT systems and the related integration challenges as well as their placement, the de-

rived architecture includes many details and therefore many optional parts.

Let us first have a look at the overall structure of the detailed architecture. As in the overview in Figure 1, again the information flow goes from bottom to top, beginning with sensors in production (on the shop floor) or in products (somewhere out in the field). From there, sensor data moves on to a more or less Cloud-based back-end system. In principal, this can be public, private or hybrid Clouds, classical (dedicated) on premises servers or combinations (Cloud systems working together with on premises IT). That area is where the data processing takes place and where applications and services are provided that generate an added value for actuators and human users. In the detailed architecture, a fourth area is added where additional data might be included from sources in the web or internet. The latter can be seen as the classic internet of information and services in contrast to the (industrial) internet of things, which adds lots of sensor devices and corresponding data. On top, user interfaces can be found – either for human users or for actuators. These may simply be remote controllable products such as independent vehicle or heaters. In a more complex scenario, machines in production that can be remotely configured to produce different variants of products give additional examples — depending on automatised decisions made by back end services in the Cloud.

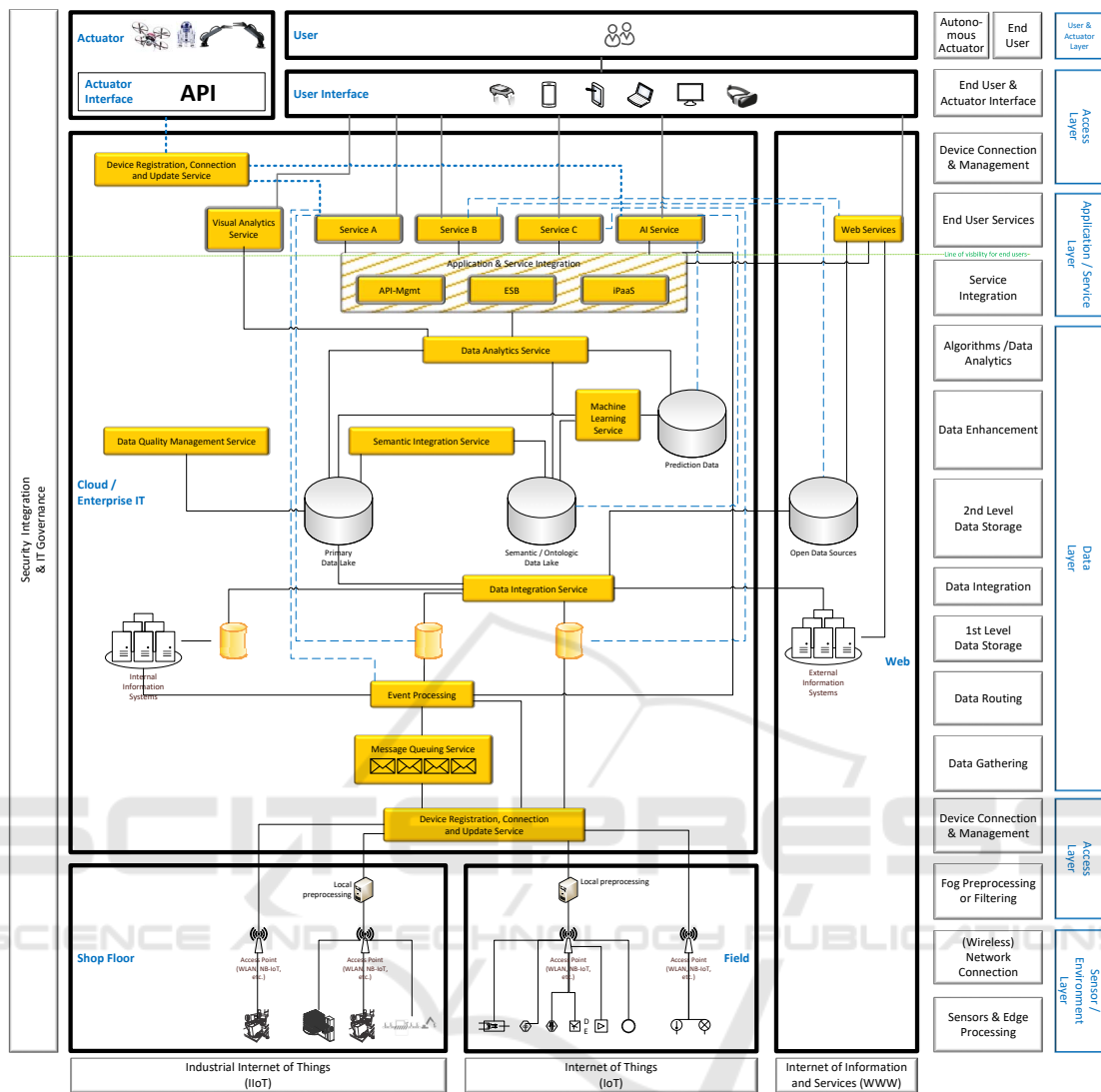


Figure 2: Detailed overview of common Cloud and IoT system components for demonstrating possible integration challenge occurrences.

To the left and right of these core areas describing the information flow, boxes describing the different integration tasks and levels can be found, such as device connection or service integration. To the far right, we displayed the basic layers of integration equal to the Overview in Figure 1.

Figure 2 adds details especially in the Data Layer. Data needs to be gathered either by putting it into some sort of message queue or by routing it into a database directly. One may have event processing included, which might also use information from other internal information systems such as Enterprise Resource Planning (ERP) or Production Life-cycle Management (PLM) systems. When data is collected in the first level data storage, it may be integrated into a data lake (2nd level storage) by some data integra-

tion service. This data integration service may also connect with data sources from the internet. Examples may be external information systems, e.g. from suppliers, web services like weather forecasts, real-time traffic information or open data sources of all kinds. The next possible step is the enhancement of the data in one or several (distributed) databases. Depending on the size and usage, these are also called data lake. The enhancement can be done by managing data quality, by semantic integration or by letting machine learning services create new information based on the second level data storage to give some examples. On top of that, the actual analysis starts, making use of the aggregated and enhanced data and finally completing the Data Layer.

Within the Application Layer all the services that

are finally processing and using the data in order to offer it to human users or in order to influence or control actuators in machines or products can be found. These services range from visual analytics services for human data scientists to artificial intelligence services that may be the foundation of digital assistants and new forms of customer services. Also they may be the classics of enterprise systems from office, from customer relationship management (CRM) to enterprise content management (ECM) systems and the likes. Of course information needs to be passed between those application services and therefore the Application Layer also needs to comprise an application & service integration component, such as API management services, an enterprise service bus (ESB) or an iPaaS which is basically an ESB as a service from the (public) Cloud.

Below the Data Layer as well as on top of the application layer we find an Access Layer. The Access Layers connect the sensor and user/actuator areas with the IT System area. Of course, actuators and sensors are often part of the same device or production machine, and the data and commands also may have to go all the way back through Edge or Fog processing and network connection in order to reach the actuators. However, these two are separated from each other for the purpose of better comprehension. In the Access Layers the integration of networked sensors, actuators and finally also human users with the (possibly Cloud-based) IT back-end takes place. Some sort of device management may be used here and in the case of Edge or Fog computing also local data pre-processing may take place here. The upper Access Layer provides human access to services (and the data included) or integrates actuators with the IT system through an actuator API.

The task of integration spans throughout the whole picture and due to the complexity of the topic, it will not be described in detail in this architectural overview. All the same, security integration may not be underestimated. It comprises many different fields, starting from technical aspects such as identity management and over-the-air (OTA) firmware updates for sensors and actuators throughout security processes and incident response and not ending with the setup of service level agreements or the encryption of data flows.

Whereas the architecture contains six layers, we discuss only the first four rather technical layers in detail for the integration challenges, since challenges on the last two layers are usually covered by the Access Layer or are part of the application development instead of the overall system setup. These are the Sensor/Environment Layer, the Access Layer, the Data

Layer and the Application Layer.

3 INTEGRATION CHALLENGES

Within this section, for the four technical layers of our architecture which are most relevant for the integration, challenges are depicted and described.

While most challenges are located predominantly at one layer, security is a challenge class which ranges over all layers and is therefore not listed for the layers. As more and more systems are connected with each other the overall vulnerability increases dramatically (van Velzen, 2017). The challenges affect nearly every component and include on-boarding, (single) sign on, (distributed) identity management, access delegation, encryption, network separation, incident management and many more. Since security is a topic requiring investigation on a different level, we exclude it from our list of challenges. We included it here and in the architecture to respect it when considering complex systems, but to really take security challenges and their effect to systems implementation into account, a separate survey is necessary.

3.1 Environment Integration

The challenges in the Environment Layer are about setting up the devices in their physical environment as well as providing the basic network functionality.

- **Device Placement.** The placement of the devices can be as easy as just placing them somewhere, but can also be a complex task. Depending on the technical requirements of the devices themselves (e.g. being waterproof) and their tasks, the environment has to be chosen appropriately (e.g. a temperature sensor for heating control behaves drastically different when placed interior, exterior or in sun affected places). For Smart Products and wearables, the sensor integration into a product can be a major task, such as embedding them into clothing (Chen et al., 2017).
- **Mobility.** Whereas many protocols and devices are designed to be used stationary, mobility such as for smartphones and wearables brings new challenges. Besides of the power supply, adequate network protocols have to be used and the devices have to be robust and precise under movement (Pereira et al., 2013) (Botta et al., 2014).
- **Network Access.** While a broad availability network is quite common, especially for enterprises, there are still areas where the necessary bandwidth or even internet at all is an issue (Clark, 2018).

In addition, the environment can contain disruptive factors limiting the possible connection types (Apple Inc., 2017) which can be the case e.g. in manufacturing halls.

- **Power Supply.** The energy consumption of the devices varies depending on the task they are used for and their frequency. Whereas for many stationary devices a continuous power supply is no problem, for mobile devices and wearables this is one of the primary challenges (Partin-Vaisband, 2017). Not only physical actions, but also reading sensor values and sending data over a network connection consumes a lot of energy.

3.2 Access Integration

- **Protocols.** Whereas there are some standards such as MQTT or HTTP (REST) establishing, there are also many other possible protocols which can cause much effort to integrate into the IoT system (Čolaković and Hadžialić, 2018). For machine to machine communication, OPC UA (OPC Unified Architecture) is a common and important protocol, but for higher level applications such as monitoring, the protocol compliance may be not given (Hazarika et al., 2015).
- **Gateways.** In complex IoT systems, gateways can play an important role acting as message brokers. A message broker provides communication channels for both things and a platform, which helps to improve the scalability and reliability of IoT systems (Pipatsakulroj et al., 2017). Well designed message broker systems can drastically increase the overall system robustness and prevent or reduce the single points of failure. They also help decoupling different system components (Kolluru and Mantha, 2013), fostering the implementation of SOA or microservice architectures. IoT smart gateways may improve the performance of IoT devices inserted into an IP backbone even in industrial environments without compromising the network load (Diaz-Cacho et al., 2015). Gateways can also operate as an additional abstraction layer between the devices and the rest of the system, i.e. as adapters making the system device (and protocol) independent (Busemann et al., 2012). That addresses the heterogeneous devices and components, which is one of the major challenges for IoT (Guo et al., 2018).
- **Device Management.** Besides registration and convenient configuration, there are also other device related requirements to facilitate service enabling, including software/firmware update, re-

mote diagnostics, and device capability provisioning. These are requirements usually handled by a unified device management mechanism (Ma et al., 2008).

- **Servers and Virtualisation.** When not only consuming public Cloud resources, the back end infrastructure has to be managed and provided. Dedicated servers and Cloud systems even more require extensive installation, configuration and administration. In case of (partially) self managing Cloud systems, core Cloud management such as rapid provisioning, resource changing, monitoring and reporting has to be solved (Liu et al., 2011). Powerful tools like VMWare, Cloud Foundry or OpenStack can be used to handle these challenges, but installation and configuration is not always easy, e.g. half of the enterprises that tried to implement an OpenStack, failed (SUSE, 2016).

3.3 Data Integration

- **Data Quality Management.** Decisions based on the analysis and evaluation of data and information are more likely to produce desired results, but in order to achieve that, the quality has to be ensured, i.e. the data has to be correct, reliable and complete (ISO, 2015) (Ardagna et al., 2018). In addition, the risk of bad quality of data increases with the amount and complexity of the collected data (Watts et al., 2009). When considering IoT systems such as smart home/building, a crucial challenge is to handle the weakest links such as defects or misconfiguration of sensors (Banerjee and Sheth, 2017).
- **Deletions.** Depending on the amount of collected data, storage can cause scaling challenges, at least an increase of the price. Also, not all data should be collected without question and data not creating any value can and should be deleted or not stored (Lenz et al., 2018).
- **Harmonised information model.** Different applications often use different information models (Schel et al., 2018) and the majority of global business documents are sent in formats not readable for machines (Korpela et al., 2016).
- **Rights Management.** When data is not stored separately for every application, but in shared databases or even data lakes, it might be necessary to have rights management. Features like multi-tenancy and third party managed infrastructure in Cloud environments require identity and access management mechanisms (Indu et al., 2018).
- **Analytics.** When data is used to learn and optimise or in refinements based on rules or other data,

the data has to be accessible for analytics services. Many enterprises, especially in the manufacturing industry, tend to have a rather diverse analytics landscape resulting in a complex and sometimes redundant and therefore inefficient system (Lenz et al., 2018). This may include several dedicated solutions from different areas such as analytics of process states, data quality, energy consumption and device or system condition.

- **Big Data.** When services make extensive use of huge data amounts, common data bases and computation architectures are not sufficient anymore. When IoT devices are involved it should be carefully considered whether a big data architecture is required or not. With an estimated number of 50 billion IoT devices in 2020, the IoT will be one of the main sources of big data with Cloud Computing enabling the computation and analysis of the data (Botta et al., 2014). A common architecture for handling big data is called Lambda Architecture and has two different streams, a long processing line for the long computation and a real time processing stream (Gribaudo et al., 2018).

3.4 Application Integration

Integration challenges in the Application Layer cover integration directly between applications, integration by service buses and integration of services in internet based service platforms.

- **Application Communication.** When different applications have to work with the same data or are part of the same (business) processes, the communication between them has to be solved (Schel et al., 2018).
- **Monitoring.** Due to the continuously growing complexity of infrastructure, services and systems especially in the Cloud context, effective, efficient and constant monitoring is required for proper operation and management (Botta et al., 2014) (Aceto et al., 2013).

3.5 Solution Approaches

- **Bilateral Service Integration (BSI).** Conceptually the most simple approach, the bilateral service integration usually is the most labour-intensive approach for application integration. For every two applications, the integration has to be solved anew if they are expected to work together. The bilateral integration of services is the traditional way and sometimes also called EAI (Enterprise Service Integration) (Pathak and Khandelwal, 2017). However, we have not chosen the term EAI, since it is

misleading due to its usage similar to the Enterprise Service Bus Approach (Thoopurani, 2014). BSI is also understood as a solution for the complexity problem of the bilateral approach (Mungrah and Cadarsaib, 2017) which is in conflict with the former usage. Nowadays, the bilateral approach is not considered a meaningful approach anymore and usually at least one of the solutions below is chosen. According to the EAI Industry Consortium, more than 70% of the BSI projects fail in some way, either by failing the deadline, the costs or the integration goal (Craggs, 2004).

- **Enterprise Service Bus.** The bus concept for the software level is a standard based communication framework allowing different software modules to communicate with each other in a centralised way providing transformation or translation of messages to allow for easy integration of legacy applications (Morariu et al., 2012). Service buses can have more specific forms to meet the requirements of specific industry sectors, e.g. as Manufacturing Service Buses (Schel et al., 2018). Due to the translation and transformation capabilities of service buses, the integration of them usually also covers challenges from the Data Layer.
- **iPaaS.** Integration Platform as a Service contains a set of integration services providing elastic scalability Cloud platform integration, also between different deployment models (e.g. on premises, Cloud) (Palanimalai and Paramasivam, 2015). This can be seen as an enterprise service bus for Cloud solutions which is itself consumed from a Cloud, that is, as a service. Such setup includes not only the execution of the primary business processes on third-party Cloud systems but the interactions with the other Cloud and on-premise systems on behalf of a client, as well (Suzic, 2016).
- **Service Platform Integration.** Over the last years many internet service platforms came up. These can not only act as marketing and sales channels for software services, but also add value by providing an integrated environment with middleware such as billing, single-sign on, authentication, authorisation, developer support and many more and they grow in value proportional to the platform ecosystem (Mineraud et al., 2016). Often such platforms also solve many challenges from the other layers such as data handling, analytics and even device management (Rauen et al., 2018) as well as analytics or machine learning (Díaz et al., 2016). These values can be used to solve many integration challenges at once, whereas the services have to be integrated to the platform itself according to the interfaces, standards and protocols defined by the ser-

vice platform. The main difference compared to iPaaS solutions is, that the offered services are already integrated, but the customer is limited to the platform repertoire.

4 CONCLUSION

In this paper, we surveyed, briefly presented and discussed several Cloud and IoT architectures with a focus on components relevant for integration. We also surveyed integration challenges for Cloud and IoT and derived an integration architecture as an overview of possible integration occurrences. The components are structured in six layers, namely the Environment, Access, Data, Application, User Access and User Layer, whereat the first four layers are the important ones for the technical integration. The identified challenges are assigned to the most related layers and discussed respectively. Whereat there is much literature about both, architectures and integration, to the best of our knowledge, no extensive overview existed so far. We also present the most common solution approaches bilateral service integration, enterprise service buses, iPaaS and service platforms, each covering many of the integration challenges.

REFERENCES

- Aceto, G., Botta, A., de Donato, W., and Pescapè, A. (2013). Cloud monitoring: A survey. *Computer Networks*, 57(9):2093–2115.
- Apple Inc. (2017). Potential sources of Wi-Fi and Bluetooth interference.
- Ardagna, D., Cappiello, C., Samá, W., and Vitali, M. (2018). Context-aware data quality assessment for big data. *Future Generation Computer Systems*, 89:548–562.
- Banerjee, T. and Sheth, A. (2017). IoT Quality Control for Data and Application Needs. *IEEE Intelligent Systems*, 32(2):68–73.
- Bolloju, N. and Murugesan, S. (2012). Cloud-based B2B Systems Integration for Small-and-Medium-sized Enterprises. *Proceedings of the International Conference on Advances in Computing, Communications and Informatics*.
- Bonomi, F., Milito, R., Zhu, J., and Addepalli, S. (2012). Fog Computing and Its Role in the Internet of Things. *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*.
- Botta, A., de Donato, W., Persico, V., and Pescapè, A. (2014). On the Integration of Cloud Computing and Internet of Things. In *2014 International Conference on Future Internet of Things and Cloud*, pages 23–30. IEEE.
- Busemann, C., Gazis, V., Gold, R., Kikiras, P., Kovacevic, A., Leonardi, A., Mirkovic, J., Walther, M., and Ziekow, H. (2012). Enabling the Usage of Sensor Networks with Service-Oriented Architectures. *Proceedings of the 7th International Workshop on Middleware Tools, Services and Run-Time Support for Sensor Networks*.
- Cavalcante, E., Pereira, J., Alves, M. P., Maia, P., Moura, R., Batista, T., Delicato, F. C., and Pires, P. F. (2016). On the interplay of Internet of Things and Cloud Computing: A systematic mapping study. *Computer Communications*, 89-90:17–33.
- Chen, M., Ma, Y., Li, Y., Di Wu, Zhang, Y., and Youn, C.-H. (2017). Wearable 2.0: Enabling Human-Cloud Integration in Next Generation Healthcare Systems. *IEEE Communications Magazine*, 55(1):54–61.
- Chryssolouris, G., Georgoulas, K., and Michalos, G. (2012). Production Systems Flexibility: Theory and Practice. *IFAC Proceedings Volumes*, 45(6):15–21.
- Clark, J. (2018). Bandwidth Limitations and the Cloud: The Data Center Journal.
- Čolaković, A. and Hadžialić, M. (2018). Internet of Things (IoT): A Review of Enabling Technologies, Challenges, and Open Research Issues. *Computer Networks*.
- Craggs, S. (2004). Avoiding EAI Disasters.
- Danezis, G., Domingo-Ferrer, J., Hansen, M., Hoepman, J.-H., Le Métayer, D., Tirta, R., and Schiffner, S. (2014). Privacy and Data Protection by Design – from policy to engineering. *European Union Agency for Network and Information Security*.
- Díaz, M., Martín, C., and Rubio, B. (2016). State-of-the-art, challenges, and open issues in the integration of Internet of things and cloud computing. *Journal of Network and Computer Applications*, 67:99–117.
- Díaz-Cacho, M., Delgado, E., Falcon, P., and Barreiro, A. (2015). IoT integration on Industrial Environments. *IEEE World Conference on Factory Communication Systems*.
- Douzis, K., Sotiriadis, S., Petrakis, E. G., and Amza, C. (2018). Modular and generic IoT management on the cloud. *Future Generation Computer Systems*, 78:369–378.
- European Telecommunications Standards Institute (2013). TS 102 690 - V2.1.1 - Machine-to-Machine communications (M2M); Functional architecture.
- Gholami, M. F., Daneshgar, F., Beydoun, G., and Rabhi, F. (2017). Challenges in migrating legacy software systems to the cloud — an empirical study. *Information Systems*, 67:100–113.
- Gribaudo, M., Iacono, M., and Kiran, M. (2018). A performance modeling framework for lambda architecture based applications. *Future Generation Computer Systems*, 86:1032–1041.
- Guo, H., Ren, J., Zhang, D., Zhang, Y., and Hu, J. (2018). A scalable and manageable IoT architecture based on transparent computing. *Journal of Parallel and Distributed Computing*, 118:5–13.
- Guth, J., Breitenbücher, U., Falkenthal, M., Leymann, F., and Reinfurt, L. (2016). Comparison of IoT Platform Architectures: A Field Study based on a Reference Architecture. *2016 Cloudification of the Internet of Things*.

- Hazarika, P., Shenoy, S., Tolety, S. B., and Kalekar, N. (2015). Mobile Cloud Integration for Industrial data Interchange. *International Conference on Advances in Computing, Communications and Informatics (ICACCI)*.
- Indu, I., Anand, P. R., and Bhaskar, V. (2018). Identity and access management in cloud environment: Mechanisms and challenges. *Engineering Science and Technology, an International Journal*, 21(4):574–588.
- ISO (2015). Quality management principles.
- Kolluru, N. V. S. and Mantha, N. (2013). Cloud Integration - Strategy to Connect Applications to Cloud. *Annual IEEE India conference (INDICON)*.
- Korpela, K., Mikkonen, K., Hallikas, J., and Pynnonen, M. (2016). Digital Business Ecosystem Transformation – Towards Cloud Integration. In *2016 49th Hawaii International Conference on System Sciences (HICSS)*, pages 3959–3968. IEEE.
- Lee, D. and Lee, H. (2018). IoT service classification and clustering for integration of IoT service platforms. *The Journal of Supercomputing*, 10(2):1568.
- Lenz, J., Wuest, T., and Westkämper, E. (2018). Holistic approach to machine tool data analytics. *Journal of Manufacturing Systems*.
- Linthicum, D. (2018). Edge computing vs. fog computing: Definitions and enterprise uses.
- Liu, F., Tong, J., Mao, J., Bohn, R., Messina, J., Badger, L., and Leaf, D. (2011). *NIST Cloud Computing Reference Architecture*. National Institute of Standards and Technology, Gaithersburg, MD.
- Ma, J., Liao, J., and Zhu, X. (2008). Device Management in the IMS. *Journal of Network and Systems Management*, 16(1):46–62.
- Madaan, N., Ahad, M. A., and Sastry, S. M. (2018). Data integration in IoT ecosystem: Information linkage as a privacy threat. *Computer Law & Security Review*, 34(1):125–133.
- Mineraud, J., Mazhelis, O., Su, X., and Tarkoma, S. (2016). A gap analysis of Internet-of-Things platforms. *Computer Communications*, 89-90:5–16.
- Morariu, C., Morariu, O., Borangiu, T., and Raileanu, S. (2012). Manufacturing Service Bus Integration Model for Implementing Highly Flexible and Scalable Manufacturing Systems. *IFAC Proceedings Volumes*, 45(6):1850–1855.
- Mungrah, R. and Cadarsaib, Z. (2017). Cloud Application Integration Methodology using Enterprise Application Integration. *2017 International Conference on Infocom Technologies and Unmanned Systems (ICTUS)*.
- Palanimalai, S. and Paramasivam, I. (2015). An Enterprise Oriented View on the Cloud Integration Approaches – Hybrid Cloud and Big Data. *Procedia Computer Science*, 50:163–168.
- Partin-Vaisband, I. (2017). Automated Design of Stable Power Delivery Systems for Heterogeneous IoT Systems. In Behjat, L., Han, J., Velez, M. N., and Chen, D., editors, *Proceedings of the on Great Lakes Symposium on VLSI 2017 - GLSVLSI '17*, pages 381–386, New York, New York, USA. ACM Press.
- Pathak, R. C. and Khandelwal, P. (2017). A Model for Hybrid Cloud Integration: With a Case Study for IT Service Management (ITSM). In *2017 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*, pages 113–118. IEEE.
- Pereira, P. P., Eliasson, J., Kyusakov, R., Delsing, J., Raayatinezhad, A., and Johansson, M. (2013). Enabling Cloud Connectivity for Mobile Internet of Things Applications. In *2013 IEEE Seventh International Symposium on Service-Oriented System Engineering*, pages 518–526. IEEE.
- Pipatsakulroj, W., Visoottiviset, V., and Takano, R. (2017). muMQ: A Lightweight and Scalable MQTT Broker. *2017 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*.
- Rauen, H., Glatz, R., Schnittler, V., Peters, K., Schorak, M. H., Zollenkop, M., Lüers, M., and Becker, L. (2018). Platform Economics in Mechanical Engineering: Challenges – opportunities – courses of action. *Roland Berger GmbH*.
- Salem, R. and Abdo, A. (2016). Fixing rules for data cleaning based on conditional functional dependency. *Future Computing and Informatics Journal*, 1(1-2):10–26.
- Schel, D., Henkel, C., Stock, D., Meyer, O., Rauhöft, G., Einberger, P., Stöhr, M., Daxer, M. A., and Seidelmann, J. (2018). Manufacturing Service Bus: An Implementation. *Procedia CIRP*, 67:179–184.
- Srdjan Krco, B. P. and Carrez, F. (2014). Designing IoT Architecture(s): A European Perspective. *IEEE World Forum on Internet of Things (WF-IoT)*.
- SUSE (2016). New Research Shows OpenStack Adoption Strong, But Complexities Remain.
- Suzic, B. (2016). Securing integration of cloud services in cross-domain distributed environments. In Ossowski, S., editor, *Proceedings of the 31st Annual ACM Symposium on Applied Computing - SAC '16*, pages 398–405, New York, New York, USA. ACM Press.
- Thoma, M., Meyer, S., Sperner, K., Meissner, S., and Braun, T. (2012). On IoT-services: Survey, Classification and Enterprise Integration. In *2012 IEEE International Conference on Green Computing and Communications*, pages 257–260. IEEE.
- Thoopurani, G. (2014). Rethinking B2B business: Enterprise application integration (EAI) in the cloud.
- van Velzen, J. T. (2017). Securing the Insecurable? An overview of Security for the Internet of Things. *Datenschutz und Datensicherheit - DuD*.
- Watts, S., Shankaranarayanan, G., and Even, A. (2009). Data quality assessment in context: A cognitive perspective. *Decision Support Systems*, 48(1):202–211.
- Wu, M., Lu, T.-J., Ling, Ling, Fei-Yang, Sun, J., and Du, H.-Y. (2010). Research on the architecture of Internet of things. *3rd International Conference on Advanced Computer Theory and Engineering (ICACTE)*, 2010.