Software Architecture Solutions for the Internet of Things: A Taxonomy of Existing Solutions and Vision for the Emerging Research

Aakash Ahmad[1], Sultan Abdulaziz[2], Adwan Alanazi[3], Mohammed Nazel Alshammari[4], Mohammad Alhumaid[5]
College of Computer Science and Engineering
University of Ha'il, Ha'il, Saudi Arabia

Abstract*—Recently, Internet of Thing (IoT) systems enable an interconnection between systems, humans, and services to create an (autonomous) ecosystem of various computation-intensive things. Software architecture supports an effective modeling, specification, implementation, deployment, and maintenance of software-intensive things to engineer and operationalize IoT systems. In order to conceptualize and optimize the role of software architectures for IoTs, there is a dire need for research efforts to analyse the existing research and solutions to formulate the vision for futuristic research and development. In this research, we propose to empirically analyse and taxonomically classify the impacts of research on designing, architecting, and developing IoT-driven software systems. We have conducted a survey-based study of the existing research – investigating challenges, solutions and required futuristic efforts – on architecting IoT systems. The results of survey highlight that software architecture solutions support various research themes for IoT systems such as (i) cloud-based ecosystems, (ii) reference architectures, (ii) autonomous systems, and (iv) agent-based systems for IoT-based software. The results also indicate that any futuristic vision to architect IoT software should incorporate architectural processes, patterns, models and languages to support reusable, automated, and efficient development of IoTs. The proposed research documents structured and systemised knowledge about software architecture to develop IoT systems. Such knowledge can facilitate the researchers and developers to identify the key areas, understand the existing solution and their limitations to conceptualize and propose innovation solutions for existing and emerging challenges related to the development of IoT software.

Keywords—Software and system architecture; Internet of Things; software engineering; software engineering for IoT

I. Introduction

Internets of Things (IoTs) provide a set of technologies and their underlying infrastructure(s) that connect systems, services, human, devices and things to establish the foundation of interconnected and autonomous computing systems [1, 2]. IoT-based computing systems provide the foundation for smart cities and societies with diverse services such as smart health, autonomous transportation, home service robots and industrial automation [2]. From an implementation point of view, IoT systems represent a complex integration of software with hardware complemented by networking components that enable interconnected devices and services that collect data and later process, manage and exploit useful data for automation [1]. As a typical example, consider a robotic agent as a hardware component that interacts with different home appliances (coffee maker, wash/cleaning machine), and it can be monitored and controlled through mobile application to enable housekeeping activities such as cleaning, entertaining and home security activities [3]. One of the recent studies performed by CISCO has indicated that the role of IoTs is increasing in industry scale systems and 25 billion heterogeneous devices (approximately) are expected to be part of the IoTs by the year 2020 and further growth beyond that [4]. In the current age, IoT systems represent the potential to connect, automate and operationalize health, transportation, urban servicing, to accelerate the implementation of smart cities. To realize such potential that lies with the IoT systems, a number of challenges such as engineering, implementation, maintenance, deployment and operationalization of the IoTs must be addressed for wide spread and trustworthy adoption of IoTs [5].

From software or systems development point of view, software design and architecture provides useful abstraction to engineer and develop complex systems effectively and efficiently [6]. In the context of the IoTs, software architecture can abstract the complex and implementation specific details of hardware component and software modules with high-level system view that facilitates engineers and developers for designing, developing and evaluating IoT systems in an automated way with required human decision support [8, 9]. Specifically, the autonomous robot for service and appliances operating in a domestic environment that interact with each other (robot making the coffee or perform cleaning services) can be represented as architectural components and connectors that support programmable implementation of an IoT system that enables humans to exploit automation and home servicing based on interconnected things (i.e., service robot and appliances). This implies that existing processes, practices, patterns, languages and tools for architecture-centric development of software and systems can be effectively leveraged to architect IoT-based software [7, 8]. To support sustainable design and development of IoT software, there is a need to extend the existing solutions that address the challenges for existing and emerging solutions for IoT software [10, 11, 12].

Scope and Contributions of Survey-based Study: In the most recent decade, lot of research and development efforts started and progressed on streamlining and leveraging

This research is supported by the Deanship of Scientific Research at the University of Ha'il, Saudi Arabia under the project number '160595' (Project Code: 160595)

architecture-centric principle, practices and solutions to tackle a diverse set of challenges such as implementation, deployment, standardization and compliance issue pertaining to IoT software and systems [10]. In recent years, some architectural models and reference architectures have also been proposed and implemented to unify the practices to develop IoT systems [3]. In the existing research and development, there is a gap in terms of investigating (i) what is the contribution and impact of existing research?, and (ii) what futuristic challenges are shaping up for the next generation of IoT software. In order to bridge this gap, we have used the principal of empirical research and practices of empirical software engineering for qualitative analysis of 88 that have been published in the last decade [5]. These studies represent a body of knowledge – as a catalogue of existing solutions – on architecting IoT intensive systems. We outline the primary objectives of the proposed research as:

1) Analysing the existing architectural solutions for IoTs and investigating the role of software architecture in the development of IoT-based software.

2) Classifying and comparing the existing solutions and their impacts to identify the trends of research for next generation of IoTs.

The proposed survey-based study mainly focuses on the taxonomical classification of existing research as themes to identify the prominent challenges, their recurring architectural solutions and patterns (as reusable knowledge) knowledge for IoTs.

Streamlining the past, present and futuristic research efforts that are required to develop innovative IoT solutions.

Summary of Results: The results of the study indicate a continuous growth of published research that exploits architectural models to develop IoT based software. Mobile and cloud computing technologies represent the technical backbone to implement the IoTs [19]. The analysis suggest that futuristic research and development depends on the provisioning of architectural principle, patterns, languages, tools and frameworks to effectively and efficiently develop the next generation of IoT systems. We believe that the proposed research and its results (presented as structured tables and illustrative figures) can be beneficial for (i) researchers who would like to conceptualize the state of existing research and understand needs for futuristic solutions, and (ii) practitioners who may be interested in leveraging the academic research and understanding the existing solutions and their application to industrial IoTs [9].

Section 2 provides background of software architecture in the context of IoTs. Section 3 presents research methodology. Section 4 to Section 6 present the results as research taxonomy, architectural solutions, and vision for emerging research on architecting software for IoTs. Section 7 concludes this paper.

II. Background: Software Architecture for IoTs

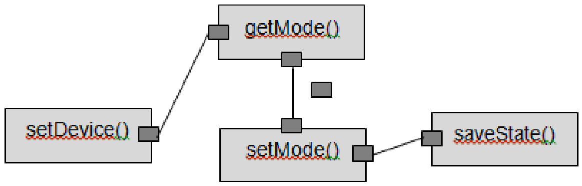
Software architecture for the IoT based systems represents a blue-print or abstraction to develop, operationalise and manage distributed and heterogeneous devices and things that communicate with each other [8, 9]. Specifically, architectural components represent the modules of source code to support the processing and data storage of the devices. Architectural connectors that interconnect the components represent interaction or message passing between the devices [6, 13]. In the following, we present Fig. 1 to show (i) a high-level structural view and (ii) operations of the software architecture for IoT systems. The notations and visualization introduced in this section helps us to discuss architectural aspects of IoT systems throughout the paper.

A. Architectural Strucure for IoT based Software Systems

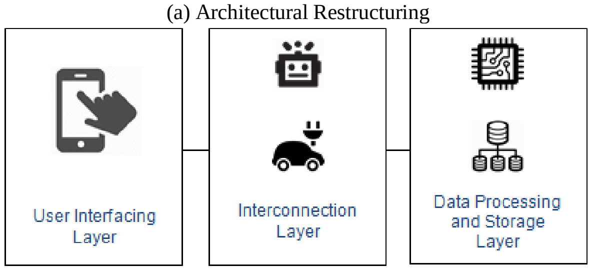
To discuss architectural structuring for IoTs, Fig. 1 presents a side by side comparative view of (i) devices and their interactions in an IoT system (Fig. 1(a)) and (ii) architectural view of the software that manages and operationalises IoT devices. The view in Fig. 1 is consistent with the presentation of software architecture in [10]. Specifically, Fig. 1(a) highlights the structure of the IoTs and their interconnections at different levels. Specifically, Fig. 1(b) presents the corresponding software architecture (components and connectors) corresponding to the IoT structure in Fig. 1(a). First, in Fig. 1(a) an IoT system is structurally divided into three distinct layers namely, *User Interfacing*, *Interconnection*, and *Data Processing and Storage* layers. Second, different architectural components and their connectors are distributed among different layers to support the IoT systems that are detailed in the next subsection. We conclude that to develop, manage, and operationalise IoT architecture, layered architecture patterns represents a suitable abstraction that can be extended and customized as per the needs of the system [11, 12].

B. Operations of the Software Architecture for IoT Systems

After the discussion about architectural structure, we now present the operations of IoTs that are supported by software architecture. Operational aspects of the IoT systems refer to the functionality performed and tasks completed by the devices (and things) that are part of an IoT system. For example, a robot that is used for home servicing is an IoT device that can interact with other IoT devices such as coffee maker or refrigerator to autonomously perform home automation services [18]. In the context of Fig. 1(a), at the User Interfacing layer the user is able to control and manipulate any IoT devices by means of their computers or portable devices. Based on user control, the IoT devices at the interconnection layer communicate with each other to perform the required tasks. Finally, at the Data Processing and Storage Layer, the data produced or consumed by IoT devices is processed and stored. Fig. 2(b) highlights the required architectural components and their interconnections that support software modules to manage and operationalise IoT system. For example, in Fig. 1(b) the architectural component named *DeviceMode* helps to configure the mode of a device such as *Start*, *Pause*, and *Shutdown*. For example, a typical scenario includes the user at home starting the home service robot (that is controlled though a mobile device) to make a cup of coffee for the user.



(a) Architectural Restructuring



(b) Layered View of the IoTs

Fig. 1. High-Level view of the Software Architecture and Components of IoT Systems.

The discussion in this section, based on the illustrations in Fig. 1 highlight that architectural components, connectors and layers (i.e., layered architecture pattern) can abstract the source code and implementation level complexities to develop, operationalise and maintain IoT-based software systems effectively. In this paper, our focus is to identify, analyse and synthesise state-of-the-art for architecting software for IoT-based systems. By presenting the architectural structure and its operationalisation helps us to discuss the effective role of software architecture and architectural notations for IoT-based systems [13, 16].

III. Research Method

In order to conduct this research, we have used the principle and procedures of the systematic literature reviews and systematic mapping studies that provides guidelines as a systematised method for collecting, analysing and reporting the data [5]. Fig. 2 presents an overview of the research methodology that is divided into three different phases namely (i) Planning the Research Study, (ii) Identifying and Synthesising the Data, (iii) Reporting the Results. In the remainder of this section, we present the research methodology, briefly summarizing each phase of Fig. 2, as detailed below. Additional information about the methodology are presented as a dedicate report in [14].

A. Phase I – Planning the Research Study

The first phase relates to planning the research study before proceeding further that comprises of following two steps.

Step 1. Identify the Needs for the Study: We need to identify and justify the needs for the proposed research study. The rationale, justification, and proposed contributions of this research study have already been detailed earlier (see Section 1). The study aims to analyse and highlight the use and effectiveness of software architecture solutions to design, engineer IoT software and systems.

Step 2. Specify the Research Questions: After defining the scope and outlining the needs, one of the most important steps is to outline the Research Questions (RQs) for the study. The RQs help to objectively assess the data and systematically present the outcomes and results of the proposed research study. Below, we outline three RQs that need to be investigated. Answers to these RQs provide the findings and results of this research study.

RQ-1: What are the existing research areas, their classification, and categorisation that support architecture-centric solutions for IoT-driven systems?

RQ-2: What challenges exist and what solutions are provided to architect software for IoT-based systems?

RQ-3: What the existing and emerging research trends and focus on architecting software for IoT-based systems?

B. Phase II – Collecting the Data for Study

After the planning, the next phase relates to collecting the data that can be analyzed to conduct the study and interpret its results. Specifically with RQs outlined, we now need to identify and synthesis the data to answer these RQs. This phase comprises of 2 steps that include identification of primary studies and collecting data from these studies. Both these steps are detailed below.

Step 1. Identification of Primary Studies: The first step in this phase relates to identification of the primary studies (as published literature). In order to identify the studies, we derive a search strategy as illustrated in Fig. 3. As in Fig. 3, we derived specific search strings to identify a total of 12390 potentially relevant studies to be reviewed. However, based on the screening (analyzing the titles and abstracts of the studies), a total of 7682 studies were short listed. Finally, based on further qualitative analysis a total of 88 studies have been selected for reviewing to support the proposed research. The list of 88 studies selected for review in this research is presented in Appendix A.

Step 2. Collection of Data for Reporting Results: After the studies have been identified (see Fig. 3), the next step involves a systematic and objective collection of the relevant data. To do so, Table I provides a structured format having multiple Data Points (DPs) to precisely capture the relevant data. The DPs have been derives based on the specified RQs. For example, in the context of RQ-1, DP-6 documents data about the challenges for architecting software IoT systems as in Table I. Table I presents a simplified view of structured data collection comprising of a total of 13 DPs.

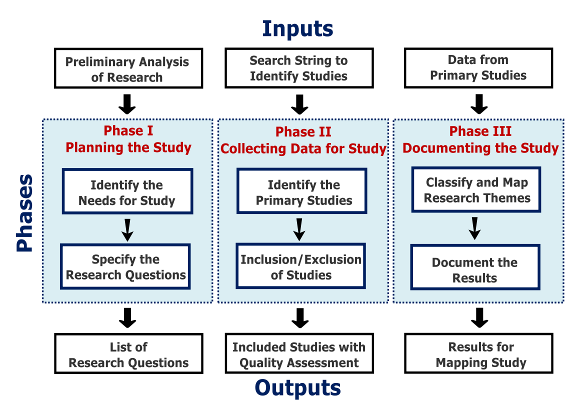


Fig. 2. Overview of the Research Method.

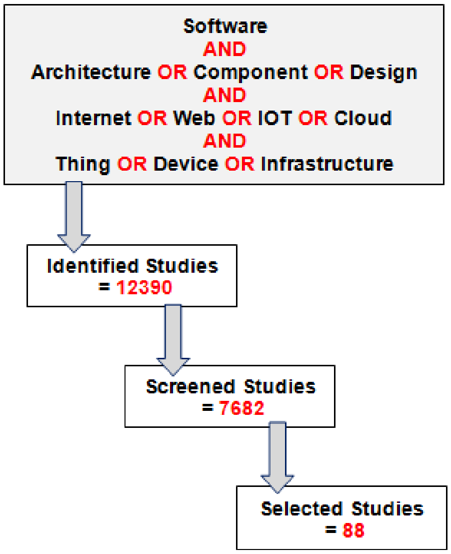


Fig. 3. Overview of the Strategy for Literature Search.

TABLE. I. STRUCTURED TEMPLATE TO DOCUMENT RELEVANT DATA

Data Point (DP)	**Description**
Generic Information	
DP-1. Study ID	Unique identification of the study
DP-2. Title	Title of the study
DP-3. Authors	List of authors
DP-4. Year of Publication	Year of publication for the study
DP-5. Study Focus	Research focus of the study
Challenges and Solutions	
DP-6. Challenge	Challenges highlighted in the study
DP-7. Proposed Solution	Proposed solution for the challenge
DP-8. Usage of Solution	Application or usage of the proposed solution
Architectural Patterns	
DP-9. Pattern Name	Name of the identified pattern
DP-10. Pattern Usage	Usage or applicability of the pattern
DP-11. Pattern View	Visual representation of the pattern
Existing and Emerging Trend	
DP-12. Existing Trends	Existing trend of research in the study
DP-13. Emerging Trend	Emerging or futuristic trend of research in study

C. Phase III – Documenting the Study to Report the Results

Finally, the last step involves documenting or reporting the results of the study. The results have been reported in dedicated sections (Section 4 to Section 6, answering RQ-1 to RQ-3) in the remainder of this paper. We conclude the discussion about research method by presenting some threats as potential issues that can affect the validity of proposed research.

Threat I – Bias in Study Identification: It relates to the inherent potential bias in the identification of most relevant studies to conduct the research. This means that relevant studies with misleading title or contents could be selected for data collection. Such bias can pose threats to the credibility of the results. In order to avoid this bias, we have tried to define relevant search strings to search and identify the most relevant studies and devised a three step criteria for the identification of the relevant studies, as in Fig. 3.

Threat II – Objective Reporting of the Results: Another threat is about objectivity of reporting the results as part of documenting the findings of the research study. This can relate to potential ambiguity in reporting the results that may lead to wrong conclusions of the research. In order minimise this threat, we have used a structured format (Table I) with defined data points to capture the relevant data. These data points are derives based on the RQs and help us to document the relevant data for each RQ individually and objectively.

The results of this research study – by answering RQ-1 to RQ-3 – are presented from Section IV to Section VI.

IV. A TAXONOMY OF SOFTWARE ARCHITECTURES FOR IOTS

We present the results for RQ-1 that focuses on outlining taxonomy and discuss the classification and categorisation of various architectural solutions for IoT software. We answer RQ-1 based on research taxonomy as illustrated in Fig. 4. Table I extends the taxonomy in Fig. 4 by presenting structured themes and sub-themes. The research taxonomy (in Fig. 4, Table I) is created by means of applying the thematic analysis and ACM classification scheme as in [17]. The taxonomy is defined as naming and hierarchical organization of existing topics and their sub-topics to outline and investigate the contributions of existing research. Both Fig. 4 and Table II support each other for a comprehensive presentation of the research taxonomy. Fig. 4 provides a blue-print as a generic view of various research themes of software architecture for IoTs.

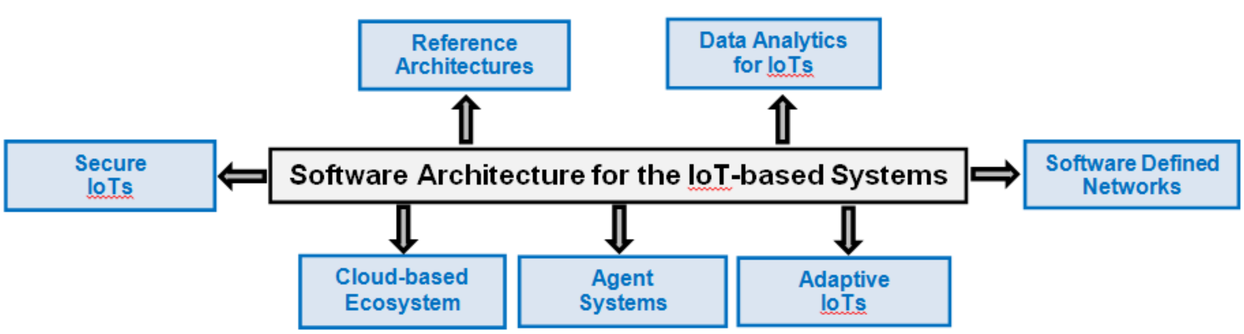


Fig. 4. A Taxonomy of Research Areas on Architecting Software for IoTs.

TABLE. II. RESEARCH AREAS, RESEARCH THEMES, NUMBER OF STUDIES AND AVAILABLE EVIDENCE

Research Area and Themes (as per Fig. 4 and Fig. 5)	**Number of Studies**	**Available Evidence (Published Studies as in Appendix A)**
1. Cloud-based Ecosystems		
2. 1.A	3	[RS-29, RS-47, RS-53]
3. 1.B	**7**	[RS-1, RS-64, RS-5, RS-9, S40, RS-42, RS-44]
4. 1.C	**5**	[RS-8, RS-18, RS-32, RS-46, RS-54]
5. 1.D	**6**	[RS-15, RS-17, RS-19, RS-34, RS-70, RS-75]
6. Software Defined Networks		
7. 2.A	**2**	[RS-12, RS-86]
8. 2.B	**3**	[RS-4, RS-23, RS-45]
9. 2.C	**2**	[RS-81, RS-25]
10. 2.D	**3**	[RS-33, RS-84, RS-88]
11. Data Analytics for IoTs		
12. 3.A	**3**	[RS-43, RS-51, RS-85]
13. 3.B	**2**	[RS-77, RS-80]
14. 3.C	**2**	[RS-66, RS-71]
15. Reference Architectures		
16. 4.A	**2**	[RS-16, RS-74]
17. 4.B	**8**	[RS-7, RS-26, RS-27, RS-28, RS-30, RS-41, RS-76, RS-87]
18. 4.C	**7**	[RS-31, RS-60, RS-63, RS-72, RS-11, RS-39, RS-73]
19. 4.D	**4**	[RS-22, RS-48, RS-62, RS-82]
20. Adaptive IoTs		
21. 5.A	**2**	[RS-38, RS-56]
22. 5.B	**5**	[RS-3, RS-13, RS-49, RS-50, RS-52]
23. 5.C	**3**	[RS-58, RS-69, RS-79]
24. 5.D	**9**	[RS-6, RS-21, RS-35, RS-36, RS-55, RS-57, RS-67, RS-68, RS-83]
25. Agents Systems		
26. 6.A	**3**	RS-10, RS-20, RS-65]
27. Secure IoTs		
28. 7.A	**2**	[RS-2, RS-78]
29. 7.B	**1**	[RS-61]
30. 7.C	**4**	[RS-14, RS-24, RS-37, RS-59]

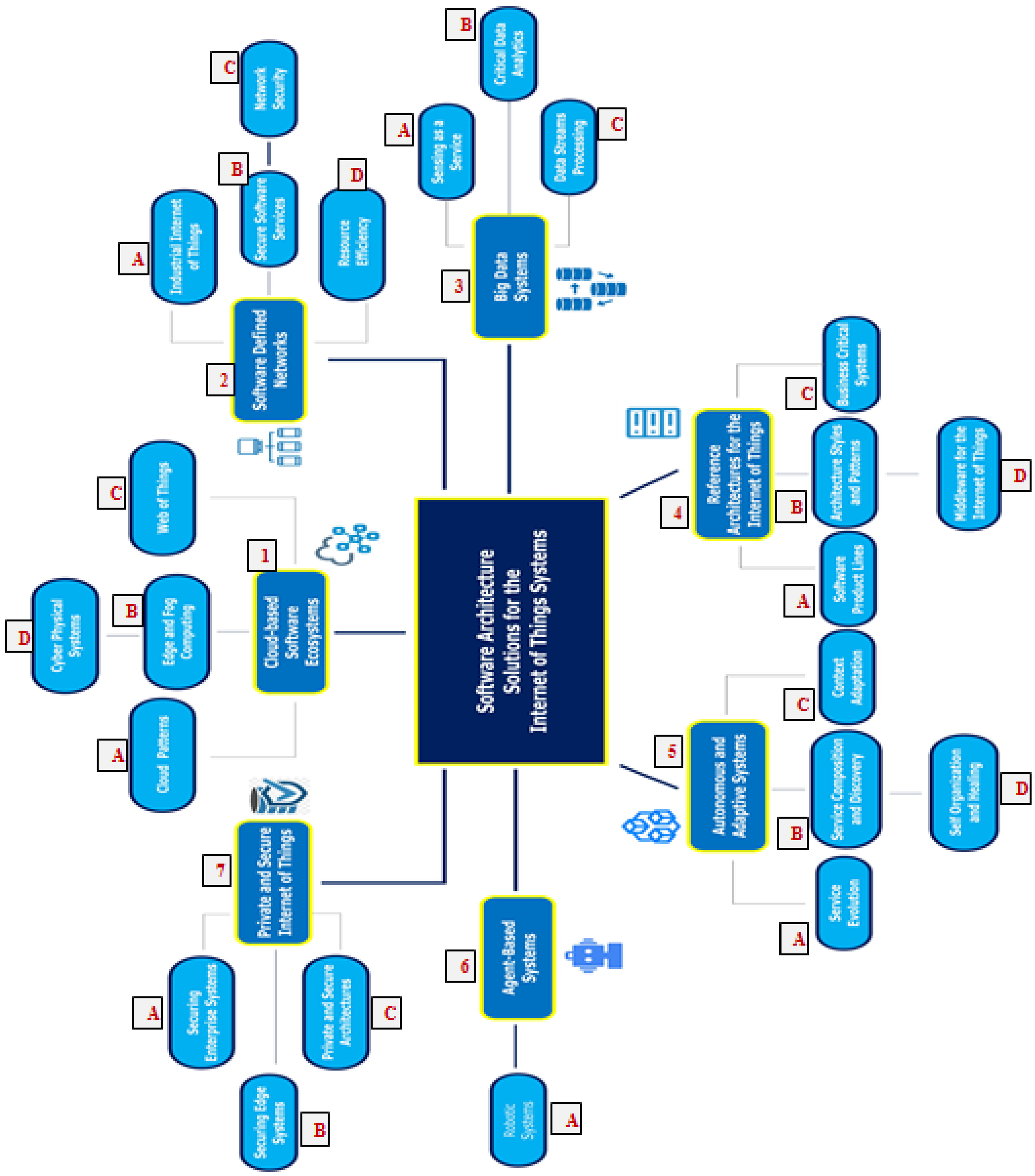


Fig. 5. Thematic Classification of Research Areas and their Research Themes.

In comparison, Table II provides structured information about the high-level research themes, sub-themes along with the references to the published studies as available evidence for each research theme and sub-theme. In Table II, we have two level organisation of the existing research named as (i) Area of Research and (ii) Research Themes. For example, as in Table I, the research study [RS-6][1] falls under the area Adaptive IoTs (i.e., Area of Research), specifically [RS-6] supports software architecture for Self-healing IoTs (i.e., Research Theme). In the remainder of these sections, based on Fig. 3 and Table II, we discuss the existing areas of research and research themes that provides us the foundation to discuss challenges and proposed architecture-centric solutions along with the emerging research discussed later in the paper.

[1] The reference [RS-n] (RS: Research Study and n is a number from 1 to 88) reflects referencing to research studies for the review as in Appendix.

A. Area of Research overviews a broader focus of the existing research and its contribution(s) in a defined scope. Fig. 4 shows that based on our thematic analysis process, we have discovered a seven areas of research (along with their sub-areas/research classification) that highlights the prominent challenges for IoTs and frequently used software architecture solutions to address the challenges for IoT systems.

B. Research Classification represents an extension and specific contributions of existing research in Table II under the generic area of research that are derived in Fig. 4. As an example, in Table II, the research studies [RS-6, RS-21, RS-35] that support software architecture for IoTs are under the research area of Adaptive IoTs. The research area of Adaptive IoTs support specific research themes such as of Self-healing IoT Systems. Our analyses have discovered 25 research themes classified under 07 areas of research to support software architecture for IoTs as detailed below. In **Table II**, the research area Cloud-based Ecosystem having 04 research themes (i) Web of Devices, (ii) Patterns for IoT Cloud, (iii) Cyber Physical Systems, and (iv) Edge Commuting.

V. Software Architecture-Centric Solutions for IoT Systems

We present the results for RQ-2 that focuses on presenting architectural solution for challenges relating to IoT systems. We have used the taxonomy of existing research (Fig. 4, RQ-1) to present the architectural solutions presented in Table III. Table III helps us to objectively present the challenges and their corresponding architecture solutions for IoTs [21]. Table III focuses on presenting (a) Challenges for IoT Systems, (b) Solutions to Challenges for Architecting IoT Systems, (c) Research Themes that support Architectural Solutions, (d) Generic View of the Software Architecture. To collect the data mentioned above, we used Table II (see Section 3) to capture the required information (Challenges and Solution (DP-6 to DP-8). We exemplify one of the challenges and solutions as detailed in Table III.

1) Challenges: How to engineer and develop a heterogeneous eco-system of cloud computing systems to support as a service model for infrastructure/platform/ software?

2) Architectural solutions for IoTs: The architectural solution provides a mediator that helps a seamless integration between various heterogeneous cloud providers to support an eco-system of clouds. However, the communication between the clouds is maintained through a mediator that involves communication latency in the ecosystem.

3) Research themes: The research themes include (a) Edge Computing [RS-1, RS-5, RS-64,], (b) Web of Devices [RS-8, RS-18, RS-32], (c) Cyber Physical System [RS-15, RS-17, RS-19], (d) Patterns for IoT Cloud [RS-29, RS-47, RS-53]

4) Architectural view: It is presented as '*Software Architecture for Cloud Ecosystem*' in Fig. 6(a).

VI. Emerging Research on Architecture-based Solutions for IoT Software

The answer to RQ-3 focuses on streamlining the emerging trends of research as potential areas of emerging research for IoT systems and software. We summarise the most relevant emerging trends that could be further explored and researched to devise state-of-the-art solutions for architecting IoT software. Based on the published research in the last 03 years (2017 to 2019), we have identified the following research themes as emerging topics with their potential as futuristic challenges and innovative solutions to engineer IoT-intensive systems.

Trend I: Architecture-centric Languages and Reusable Patterns.

In the architectural context, architectural languages and reusable patterns they support provide the necessary structure, semantics, and reusable knowledge for architecture-based modeling, development and management of software systems. Our analysis suggests that in recent years, architectural languages are being investigated as structured mechanism to exploit reusable specification and knowledge for architecture-centric development of the IoT systems. The primary challenge with architectural languages and patterns is that they are derived from empirical knowledge that needs to be systematically discovered, structurally documented, and frequently reused for system wide usage. As part of future solution for architecting IoTs, there is a growing need for research and development to exploit the existing foundations to develop solutions such as meta-models, ontologies and pattern catalogues for architecting IoT software. Such initiatives can support innovative solutions to model, develop, evolve, operate, and manage IoT-driven systems in an effective and efficient manner [22, 23, 25].

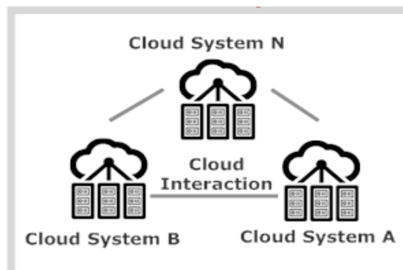
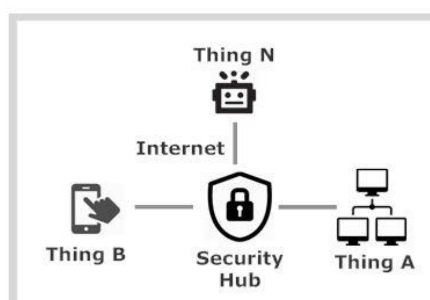
Trend II: Context-aware IoT Architectures

Context-aware computing support adaptive systems can evolve the operations and functionality of a system under changing requirements. Mobile computing is fast becoming an integrated part of the IoT systems. Mobile systems that support portability are regarded as context-aware computing nodes that can sense contextual information such as current location, time etc. to adapt a system as per contextual requirements. By means of integrating mobile computing with IoTs, we could exploit mobile-driven IoTs that are portable and context-sensitive. Specifically, context-aware IoTs can gather and process contextual information (such as geo-location, temperatures, time of the day) to adapt IoT systems in the field of robotics. The applications of context-aware IoTs can range from agent-based systems for smart city systems including but not limited to smart traffic management, smart healthcare, recommender systems and home service autonomous and adaptive agents [24]. However, in the context of mobile-driven context-aware systems, security and privacy of user data and device information are of prime importance [20].

Trend III: Reference Architectures for Industrial IoTs

In recent years, IoT systems are being increasingly adopted in industrial context ranging from industrial automation to manufacturing and supply chain. Reference architectures have proven to be useful to provide a blue-print and high-level view for the system under consideration [9, 10]. For example, the reference architecture named RAMI 4.0 supports industrial automation based on IoTs. Future research and development requires more diverse reference architectures and their standardization that could be used across industries such as manufacturing, transportation and logistics [15]. For such systems, the role of reference architectures and models become fundamental to design, develop, and manage complex IoT systems.

TABLE. III. A MAPPING EXISTING CHALLENGES AND CORRESPONDING SOLUTIONS FOR ARCHITECTING IOTS

Challenges for IoT-based Software Systems	Solutions for Architecting IoT-based Software Systems
Challenge A: Cloud-based Ecosystem	
Primary Challenge: How to support a (heterogeneous) eco-system of cloud-based software that supports (infrastructure/platform/software) as a service model offered by various cloud providers to third party cloud service subscribers?	*Proposed Solution:* The IoT based solution supports the communication among multi-clouds - the interconnected things that produce ad consume data - using a central Cloud Hub. The Cloud Hub facilitates the interaction among the different clouds to collaborate and connect as part of the ecosystem.
Research Themes with Published Evidences	**Generic View of Software Architecture**
1-a. Edge Computing Systems exploits architectural configurations to push the computation and storage of devices to the edge of the network to improve performance of the IoTs. **Available Evidence(s):** [S1, S64, S5, S9, S40, S42, S44] *1-b. Web of Things (WoT)* support potently connected web of devices and things to support universally accessible IoTs **Available Evidence(s):** [S8, S18, S32, S46, S54] *1-c. Cyber Physical Systems* represent heterogeneous and massively distributed systems to support IoTs for cyber physical systems. **Available Evidence(s):** [S15, S17, S19, S34, S70, S75] *1-d. Cloud Computing Patterns* represent generic and reusable knowledge and practices that supports reuse-driven development, maintenance and evolution of software architectures for IoT-based cloud-ecosystems. **Available Evidence(s):** [S29, S47, S53]	The architectural view presents a generic view in terms of highlighting the Cloud Hub as a mediator that helps various cloud providers to bind with each other using the mediator as illustrated in Figure 6 a.  Figure. 6 a. Overview of Generic Software Architecture for Cloud Ecosystem.
Challenge B: Secure Internet of Things	
Primary Challenge: How to enable the security and privacy of data that is produced, consumed or shared by the interconnected things (i.e., heterogeneous devices) in the IoT systems?	*Proposed Solution:* The solution introduces a Security Hub that exploits the mediator architectural pattern to mediate the communication between the connected devices/things and ensure the security of the IoT system.
Research Themes with Published Evidences	**Generic View of Software Architecture**
2-a. Security of Edge Computing supports secure communication, computation and storage of the nodes deployed as edges in IoT based architectural configurations. **Available Evidence(s):** [S61] *2-b. Secure and Private Enterprise Software* supports the security of computation infrastructures and privacy of data in industry scale enterprise software based on IoTs. **Available Evidence(s):** [S2, S78] *2-c. Reference Architecture for Secure IoT Systems* provides a reference architecture and a framework as guidelines and enabling methods to design and develop secures IoTs. **Available Evidence(s):** [S14, S24, S37, S59]	The architectural view presents the mediator, i.e., Security Hub that monitors and manages the data communication between the devices that may introduce the communication latency as illustrated in Figure. 6 b.  Figure 6.b. Overview of the Software Architecture for Secure IoTs.

(CONTD.)

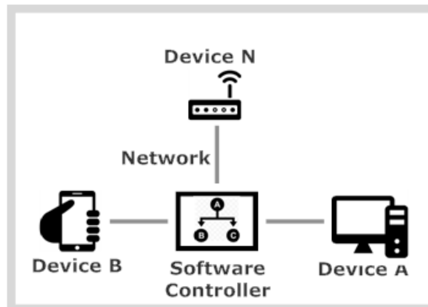
<table>
<tr><th colspan="2">Challenge C: Software Defined Networks for IoTs</th></tr>
<tr><td><u>Primary Challenge:</u> How to enable software-driven (logic-based) dynamic interconnection and coordination of the things in IoT based systems?</td><td><u>Proposed Solution</u> The proposed solutions integrate a software controller that monitors and manages the necessary logic for runtime manipulation of the interconnections in IoT systems.</td></tr>
<tr><th>Research Themes with Published Evidences</th><th>Generic View of Software Architecture</th></tr>
<tr><td>3-a. <u>Software Services in IoTs</u> supports the runtime coordination and security of software as a service in IoT based systems.
Available Evidence(s): [S4, S23, S45]</td><td rowspan="4">The architectural view presents a Software Controller that acts as an orchestrator for runtime management and configuration of the devices to support software and logic-driven networking of the IoTs as illustrated in Figure. 6 c.


Figure. 6 c. Overview of Generic Software Architecture for Software Defined Networking of IoTs</td></tr>
<tr><td>3-b. <u>Software Defined Security</u> enables software-driven monitoring and management of security for the IoT systems.
Available Evidence(s): [S81, S25]</td></tr>
<tr><td>3-c. <u>Resource Efficient IoT</u> enables software systems to support the computation, storage, energy, and communication efficiency for IoT networks.
Available Evidence(s): [S33, S84, S88]</td></tr>
<tr><td>3-d. <u>Industrial IoTs</u> is supported with dynamically configured software services that can manipulate and automate the devices in industrial IoTs.

Available Evidence(s): [S12, S86]</td></tr>
</table>

<table>
<tr><th colspan="2">Challenge D: Software Agents for IoTs</th></tr>
<tr><td><u>Primary Challenge:</u> How to exploit the devices and things in the IoT systems to engineer and develop autonomous and adaptive agents that are interconnected and web accessible?</td><td><u>Proposed Solution:</u> The proposed solutions consist of two parts, i.e., IoT System and the Interconnected Software Agents. The agents (i.e., autonomous, adaptive and distributed components) can be connected to the IoT systems to perform their tasks.</td></tr>
<tr><th>Research Themes with Published Evidences</th><th>Generic View of Software Architecture</th></tr>
<tr><td>4-a. <u>Software Agents</u> can support a collaborative network of agents that actively coordinate to complete the assigned tasks in an IoT system.

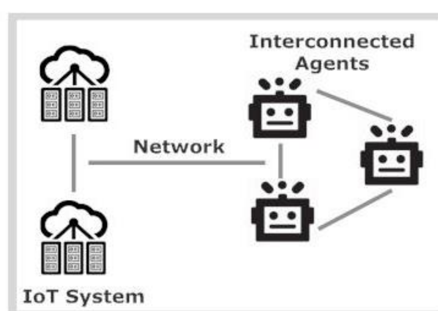
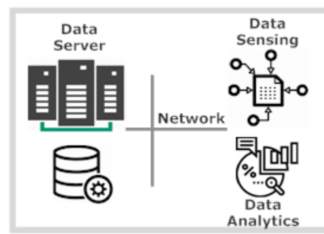
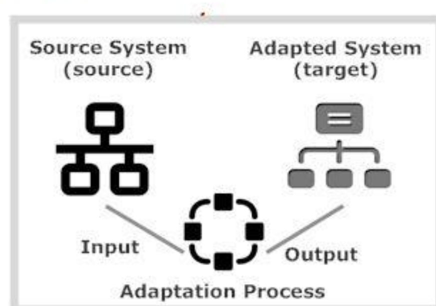
Available Evidence(s): [S10, S20, S65]</td><td>The architectural view presents network-based coordination between the IoT system and software agents as illustrated in Figure 6 d.


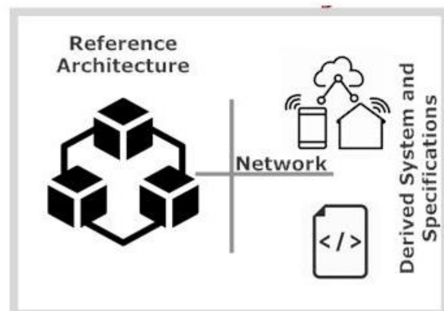
Figure. 6 d. Overview of Generic Software Architecture for Software Agents of IoTs.</td></tr>
</table>

(CONTD.)

Challenge E: Supporting IoT based Big Data Systems	
Primary Challenge: How to exploit the interconnected and distributed devices in an IoT system to sense (collect real-world) data and analyze it (process in real-time) for decision support systems?	*Proposed Solution:* The proposed solution exploits 'on the edge' deployed devices that acts as context-aware data sensors and communicate it with the backend server. The backend cloud-based servers can store and process the data with back and forth communication between front-end devices and back-end servers.
Research Themes with Published Evidences	**Generic View of Software Architecture**
5-a. Sensing as a Service: support loosely coupled and autonomous (web-) services that exploit the devices and sensors to collect the data. **Available Evidence(s):** [S43, S51, S85]	The architectural view presents the front-end (on the edge) IoT devices that sense data and communicates it with the backend server that process the data as illustrated in Figure. 6 e.
5-b. Data Stream Processing: is managed by IoT sensors that can be deployed at different network locations to collect and process live streams of data as part of real-time data analytics. **Available Evidence(s):** [S66, S71]	
5-c. Critical Data Analytics: is managed by IoT devices that are deployed 'on the edge' to sense and process critical data that includes health, context and urban analytics. **Available Evidence(s):** [S77, S80]	Figure. 6 e. Overview of Generic Software Architecture for IoT based Big Data Systems.

Challenge F: Adaptive IoTs	
Primary Challenges: How to support architectural models that enable autonomous and adaptive IoTs that dynamically adapts the structure and behavior of the IoT systems at runtime?	*Proposed Solution:* The solution supports an adaptation process based on IBM'S framework for autonomic computing that takes a (source/existing) IoT systems and then adjusts its structure and behavior to enable an adapted (target/new) IoT system.
Research Themes with Published Evidences	**Generic View of Software Architecture**
6-a. Service Composition: supports dynamic discovery and composition of the IoT software services to enable dynamic systems as per the contextual requirements. **Available Evidence(s):** [S3, S13, S49, S50, S52]	The architecture presents an adaptation process that contains the adaptation context and logic to dynamically reconfigure/adapts a source system to a target system at runtime in Figure. 6 f.
6-b. Self Healing: enables IoT software that is fault-tolerant and it can self-organize to continue its operations under continuously varying requirements and frequent maintenance and evolution. **Available Evidence(s):** [S6, S21, S35, S36, S55, S57, S67, S68, S83]	
6-c. Service Evolution: is supported with dynamically reconfigured services in the IoTs systems that evolve over-time as per changes in the business and technical requirements and operational environments. **Available Evidence(s):** [S38, S56]	Figure. 6 f. Overview of Generic Software Architecture for Adaptive IoTs.

(CONTD.)

Challenge G: Reference Architectures for IoTs	
Primary Challenge: How to establish guidelines and frameworks that provides the templates and blue-print to design and develop IoT systems for real world applications?	*Proposed Solution:* The solutions provide reference architectures as generic and high-level (abstract) solutions that can be instantiated with specialized architectures as (concrete) solutions for IoTs.
Research Themes with Published Evidences	**Generic View of Software Architecture**
7-a. *Enterprise and Business Critical Systems:* are developed based on reference architectures and process automation that support enterprise systems and production/assembly lines to enable industrial IoTs. **Available Evidence(s):** [S31, S60, S63, S72]	The architectural views present reference architectures as a reference framework that serves as a template and a collection of documented guidelines to architect and design IoT systems as in Figure 6. g.
7-b. *IoT for Ecommerce Solutions:* are supported by business process automation and reference architectures that automate those business processes to support the development and operations of IoT based e-commerce systems. **Available Evidence(s):** [S11, S39, S73]	
7-c. *Architectural Styles and Patterns for IoTs*: represent generic knowledge and best practices to model, develop, and evolve IoT architectures with enhanced reusability and efficiency. **Available Evidence(s):** [S7, S26, S27,S 28, S30, S41, S76, S87]	
7-d. *IoT-based Software Product Lines*: are supported by reference architectures that represent a generic solution to derive specialized product lines that enable Industrial IoT systems. **Available Evidence(s):** [S16, S74]	Figure. 6 g. Overview of Generic Reference Architecture for IoTs
7-e. *Smart City Architecture:* are enabled by interconnected IoT devices that act as backbones for smart and autonomous infrastructures for digitized urban services. **Available Evidence(s):** [S82]	
7-f. *IoT Middleware:* provides a technological layer that abstracts the hardware and software complexities to support IoT devices (hardware) and their corresponding (software) applications to support IoT systems. **Available Evidence(s):** [S22, S48, S62]	

VII. CONCLUSIONS OF THE STUDY

The research and development on architecting software-driven IoTs have progressed and matured for more than a decade to address challenges and proposed architectural solutions for IoTs. In this research study, we have used EBSE method to systematically identify the state of existing research and areas of future interests on software architectures in developing IoT-driven software systems. The result of our study highlights that existing research has focused on research areas such as cloud ecosystems, adaptive IoTs, secure IoTs, software defined networks, reference architectural models, agent systems and data analytics for IoTs. The trends suggest that emerging and future research needs to focus on architectural languages and reuse-based patterns, context-aware IoTs and reference architectures for IoTs in the context of industrial systems. The results of study can be beneficial for:

Researchers interesting in assessing the state of existing research and identify the areas of future research. By presenting research taxonomy of themes and a view of architectural solutions, the proposed study can support new research ideas and development. The study could help to understand: (i) challenges and architectural solution, along with (ii) emerging research to design new generation of architectural solutions for IoT software.

Practitioners could be interested in knowing the architectural artifacts in the development of industrial solutions for IoTs. Specifically, the discussion about reference models and architectures provide a foundation for solutions that could be applied to industrial IoTs.

ACKNOWLEDGMENT

This research is supported by the Deanship of Scientific Research at the University of Ha'il, Saudi Arabia under the project number '160595' (Project Code: 160595)

References

[1] Al-Fuqaha , M. Guizani , M. Mohammadi ; M. Aledhari ; M. Ayyash. "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications", IEEE Communications Surveys & Tutorials, vol. 17, no. 4, pp: 2347 – 2376, 2015.

[2] Andrea Zanella, Nicola Bui , Angelo Castellani, Lorenzo, Vangelista, Michele Zorzi. "Internet of Things for Smart Cities", IEEE Internet of Things Journal, vol. 1 , no. 1, pp. 22 – 32, 2014.

[3] J. O. Hamblen, G.M.E van Bekkum. "An Embedded Systems Laboratory to Support Rapid Prototyping of Robotics and the Internet of Things", IEEE Transactions on Education, vol. 56, no.1, pp. 121-128, 2013.

[4] D. Evans. "The Internet of Things How the Next Evolution of the Internet Is Changing Everything", CISCO White Paper, San Jose, USA, , 2011.13/11/2018/

[5] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson. "Systematic Mapping Studies in Software Engineering", 12th International Conference on Evaluation and Assessment in Software Engineering, vol. 8, pp. 68–77, 2008.

[6] R. Capilla, A. Jansen, A. Tang, P. Avgeriou, M. A. Babar. "10 Years of Software Architecture Knowledge Management: Practice and Future", Journal of Systems and Software, vol 116, pp. 191–205, 2016.

[7] J. Wan, S. Tang, Z. Shu, D. Li, S. Wang, M. Imran, and A. V. Vasilakos. "Software-defined Industrial Internet of Things in the Context of Industry 4.0", IEEE Sensors Journal, vol. 16, no. 20, pp. 7373–7380, 2016.

[8] P. Fremantle, "A Reference Architecture for the Internet of Things", WSO2, White Paper, [Online]. Available: http://wso2.com, 2014

[9] E. Cavalcante, M.P. Alves, T. Batista, F.C. Delicato, P.F. Pires. "An Analysis of Reference Architectures for the Internet of Things", 1st International Workshop on Exploring Component based Techniques for Constructing Reference Architectures, 2015.

[10] M. Weyrich and C. Ebert. Reference Architectures for the Internet of Things. In IEEE Software, vol. 33, no. 1, pp. 112–116, 2016.

[11] G. Campeanu. "A Mapping Study on Microservice Architectures of Internet of Things and Cloud Computing Solutions", 7th Mediterranean Conference on Embedded Computing (MECO). IEEE, 2018.

[12] M. Henry, and M. T. Moghaddam. "IoT Architectural Styles". European Conference on Software Architecture. Springer, 2018.

[13] E. Croes. "Software Architectural Styles in the Internet of Things", MS Thesis, Radboud University Nijmegen, August 2015.

[14] A. Alreshidi, A. Ahmad. "Protocol for Systematic Mapping Study on Software Architecture for the Internet of Things", Technical Report, College of Computer Science and Engineering, University of Ha'il, pp. 1 – 13, 2019. https://ahmadaakash.wixsite.com/aakash/sa4iot-ressearchprotocol

[15] Internet of Things Related Standards. IEEE Standards Association [Online:] https://standards.ieee.org/initiatives/iot/stds.html, Accessed 02/02/2019

[16] F. Ciccozzi, R. Spalazzese. "MDE4IoT: Supporting the Internet of Things with Model-Driven Engineering", International Symposium on Intelligent and Distributed Computing. Springer, 2016.

[17] R. E. Boyatzis. "Transforming Qualitative Information: Thematic Analysis and Code Development". Sage, 1998.

[18] A. Ahmad, M. A. Babar. Software Architectures for Robotic Systems: A Systematic Mapping Study. In the Journal of Systems and Software, vol. 122, pp: 16 – 39, 2016.

[19] I. Alseadon, A. Ahmad, A. Alkhalil, K. Sultan. Migration of Existing Software Systems to Mobile Computing Platforms: A Systematic Mapping Study. In Frontiers of Computer Science, Springer, 2019. https://doi.org/10.1007/s11704-019-8166-5

[20] M. Sajjad, A. A. Abbasi, A. Malik, A. B. Altamimi, and I. M. Alseadoon. Classification and Mapping of Adaptive Security for Mobile Computing. In IEEE Transactions on Emerging Topics in Computing, IEEE, 2018. 10.1109/TETC.2018.2791459

[21] A. Alreshidi, A. Ahmad. Architecting Software for the Internet of Thing Based Systems. In Future Internet, MDPI. Future Internet 2019, vol. 11, no. 7), 153; https://doi.org/10.3390/fi11070153

[22] A. Ahmad, A. W. Malik, A. Alreshidi, W. Khan, and M. Sajjad. Adaptive Security for Self-Protection of Mobile Computing Devices. In Mobile Networks and Applications (2019), Springer, 2019.

[23] A. Ahmad and M. A. Babar, "A framework for architecture-driven migration of legacy systems to cloud-enabled software," in Proceedings of the WICSA 2014 Companion Volume. ACM, 2014.

[24] A. Khan, A. Ahmad, A. Rahman, A. Alkhalil . A Framework for the A Mobile Cloud Framework for Context-Aware and Portable Recommender System for Smart Markets. In Smart Infrastructure and Applications. EAI/Springer Innovations in Communication and Computing. Springer, 2019.

[25] A. Ahmad, I. Alseadoon, A. Alkhalil, K. Sultan. Evolution of Legacy Software Towards Context-aware and Portable Mobile Computing Applications. In Proceedings of the International Conference on Software Engineering Research and Practice (SERP), pp. 3-9, 2019.

APPENDIX A. LIST OF STUDIES REVIEWED FOR DATA COLLECTION

Study ID	Title of the Study	Publication Type	Publication Year
[RS-1]	Cloud Ecosystems Support for Internet of Things and DevOps Using Patterns	Conference	2016
[RS-2]	Internet of Things (IoT): Security challenges, business opportunities & reference architecture for E-commerce	Conference	2015
[RS-3]	A Taxonomy of IoT Client Architectures	Journal	2018
[RS-4]	An Open Internet of Things System Architecture based on Software-defined Device	Journal	2018
[RS-5]	Responsive Data Architecture for the Internet of Things	Journal	2016
[RS-6]	Architecting Emergent Configurations in the Internet of Things	Conference	2017
[RS-7]	Software architecture pattern selection model for Internet of Things based systems	Journal	2018
[RS-8]	A Software Architecture Enabling the Web of Things	Journal	2015
[RS-9]	Application of cloud computing in the emergency scheduling architecture of the Internet of Things	Conference	2015
[RS-10]	Modeling landing control system of carrier-based aircraft on Internet of Things	Conference	2014
[RS-11]	Digital Enterprise Architecture - Transformation for the Internet of Things	Conference	2015
[RS-12]	Adaptive Transmission Optimization in SDN-Based Industrial Internet of Things With Edge Computing	Journal	2018
[RS-13]	A Scalable and Self-Configuring Architecture for Service Discovery in the Internet of Things	Journal	2014
[RS-14]	A Novel Secure Architecture for the Internet of Things	Conference	2011
[RS-15]	Designing a Cyber-Physical Cloud Computing Architecture	Journal	2015
[RS-16]	Building a Framework for Internet of Things and Cloud Computing	Conference	2014
[RS-17]	Software Abstractions for Component Interaction in the Internet of Things	Journal	2016
[RS-18]	From the Internet of Things to the web of things — enabling by sensing as-a service	Conference	2016
[RS-19]	A Component Architecture for the Internet of Things	Journal	2018
[RS-20]	Infrastructure Management Support in a Multi-agent Architecture for Internet of Things	Symposium	2014
[RS-21]	Software architecture of self-organizing systems-of-systems for the Internet-of-Things with SosADL	Conference	2017
[RS-22]	A Reference Model for Internet of Things Middleware	Journal	2018
[RS-23]	A SDN-based architecture for horizontal Internet of Things services	Conference	2016
[RS-24]	A Multiple Layer Sercurity Architecture for Internet of Things into MVC Design	Conference	2018
[RS-25]	An hybrid and proactive architecture based on SDN for Internet of Things	Conference	2017
[RS-26]	Design and Implementation of a Hardware Versatile Publish-Subscribe Architecture for the Internet of Things	Journal	2018
[RS-27]	T-REST: An Open-Enabled Architectural Style for the Internet of Things	Journal	2018
[RS-28]	A Layered Protocol Architecture for Scalable Innovation and Identification of Network Economic Synergies in the Internet of Things	Conference	2016
[RS-29]	IoT gateway for smart metering in electrical power systems - software architecture	Conference	2017
[RS-30]	The Internet of Things (IoT): A Study of Architectural Elements	Conference	2017
[RS-31]	Research and Application of Manufacturing Recourse Management in Manufacturing Enterprise Based on Internet of Things	Conference	2017
[RS-32]	An Avatar Architecture for the Web of Things	Journal	2015
[RS-33]	Graph Based Clustering for Two-Tier Architecture in Internet of Things	Conference	2016
[RS-34]	A novel clock synchronization architecture for IoT access system	Conference	2016
[RS-35]	Self-Organised Middleware Architecture for the Internet-of-Things	Conference	2013
[RS-36]	A Commitment-Based Approach to Realize Emergent Configurations in the Internet of Things	Workshop	2017
[RS-37]	A Reference Architecture for Improving Security and Privacy in Internet of Things Applications	Conference	2014
[RS-38]	Complex Event Recognition Notification Methodology for Uncertain IoT Systems Based on Micro-Service Architecture	Conference	2018
[RS-39]	Research on intelligent supermarket architecture based on the Internet of Things technology	Symposium	2014
[RS-40]	CEFIoT: A fault-tolerant IoT architecture for edge and cloud	Forum	2018
[RS-41]	Microservices approach for the internet of things	Conference	2016
[RS-42]	Moving Application Logic from the Firmware to the Cloud: Towards the Thin Server Architecture for the Internet of Things	Conference	2012
[RS-43]	An Architecture to Support the Collection of Big Data in the Internet of Things	Conference	2014
[RS-44]	Edge computing enabling the Internet of Things	Forum	2015
[RS-45]	A distributed software-defined multi-agent architecture for unifying IoT applications	Conference	2017
[RS-46]	A resource oriented architecture for the Web of Things	Conference	2012

[RS-47]	IoT Mashup as a Service: Cloud-Based Mashup Service for the Internet of Things	Conference	2013
[RS-48]	A Reference Architecture for federating IoT infrastructures supporting semantic interoperability	Conference	2017
[RS-49]	Automatic Generation of Distributed Run-Time Infrastructure for Internet of Things	Workshop	2017
[RS-50]	Event-Aware Framework for Dynamic Services Discovery and Selection in the Context of Ambient Intelligence and Internet of Things	Journal	2016
[RS-51]	A Holistic Architecture for the Internet of Things, Sensing Services and Big Data	Conference	2013
[RS-52]	A Decentralized Locality-Preserving Context-Aware Service Discovery Framework for Internet of Things	Conference	2015
[RS-53]	Architecture and measured characteristics of a cloud based internet of things	Conference	2012
[RS-54]	CoTWare: A Cloud of Things Middleware	Workshop	2017
[RS-55]	Self-Healing for Distributed Workflows in the Internet of Things	Workshop	2017
[RS-56]	A Notification Management Architecture for Service Co-evolution in the Internet of Things	Symposium	2016
[RS-57]	A management architectural pattern for adaptation system in Internet of Things	Conference	2016
[RS-58]	A message broker based architecture for context aware IoT application development	Conference	2017
[RS-59]	Thingtegrity: A Scalable Trusted Computing Architecture for the Internet of Things	Conference	2016
[RS-60]	A Reference Separation Architecture for Mixed-Criticality Medical and IoT Devices	Conference	2017
[RS-61]	An Architectural Mechanism for Resilient IoT Services	Workshop	2017
[RS-62]	A policy-based coordination architecture for distributed complex event processing in the internet of things: doctoral symposium	Conference	2016
[RS-63]	The design and development of an intelligent tutoring system as a part of the architecture of internet of things (IoT)	Conference	2017
[RS-64]	A new security middleware architecture based on fog computing and cloud to support IoT constrained devices	Conference	2017
[RS-65]	Towards a Reference Architecture for Swarm Intelligence-Based Internet of Things	Conference	2017
[RS-66]	A Graph-Based Cloud Architecture for Big Stream Real-Time Applications in the Internet of Things	Conference	2014
[RS-67]	Applying Architecture-Based Adaptation to Automate the Management of Internet-of-Things	Conference	2018
[RS-68]	Formally Describing Self-organizing Architectures for Systems-of-Systems on the Internet-of-Things	Conference	2018
[RS-69]	A Framework of Adaptive Interaction Support in Cloud-Based Internet of Things (IoT) Environment	Conference	2014
[RS-70]	An Architecture for Interoperable IoT Ecosystems	Workshop	2016
[RS-71]	An Open-Source Cloud Architecture for Big Stream IoT Applications	Conference	2015
[RS-72]	An Integrated Architecture for IoT-Aware Business Process Execution	Conference	2018
[RS-73]	Integrating Traditional Stores and e-Commerce into a Multi-tiered Recommender System Architecture Supported by IoT	Conference	2017
[RS-74]	Towards Defining Families of Systems in IoT: Logical Architectures with Variation Points	Summit	2015
[RS-75]	Architecture and Scheduling Method of Cloud Video Surveillance System Based on IoT	Conference	2015
[RS-76]	The Internet of Things: Insights into the building blocks, component interactions, and architecture layers	Conference	2018
[RS-77]	An energy-efficient internet of things (IoT) architecture for preventive conservation of cultural heritage	Journal	2018
[RS-78]	SecureSense: End-to-end secure communication architecture for the cloud-connected Internet of Things	Journal	2017
[RS-79]	COLLECT: COLLaborativE ConText-aware service oriented architecture for intelligent decision-making in the Internet of Things	Journal	2017
[RS-80]	A new architecture of Internet of Things and big data ecosystem for secured smart healthcare monitoring and alerting system	Journal	2018
[RS-81]	New Security Architecture for IoT Network	Workshop	2015
[RS-82]	Smart City Architecture and its Applications Based on IoT	Journal	2018
[RS-83]	Software architecture of self-organizing systems-of-systems for the Internet-of-Things with SosADL	Journal	2017
[RS-84]	Evaluating energy efficiency of Internet of Things software architecture based on reusable software components	Journal	2017
[RS-85]	Software-defined wireless network architectures for the Internet-of-Things	Conference	2015
[RS-86]	A System Architecture for Software-Defined Industrial Internet of Things	Conference	2015
[RS-87]	PMDA: A physical model driven software architecture for Internet of Things	Journal	2013
[RS-88]	EfiIoT: An efficient software architecture for internet of things	Conference	2012