



Evaluating the DevOps Reference Architecture for Multi-cloud IoT-Applications

Georges Bou Ghantous¹ · Asif Qumer Gill¹

Received: 4 November 2020 / Accepted: 11 February 2021 / Published online: 1 March 2021
© The Author(s) 2021

Abstract

DevOps originated in the context of the agile development aims to enable the continuous deployment of software applications in small releases. Most recently, organizations are taking a significant interest in adopting DevOps for automatically deploying IoT applications to multi-cloud. The interest is there; however, the challenge is how to achieve this. A new DevOps reference architecture (DRA) framework has been iteratively developed to address this challenge. This paper presents the updated DRA framework and its empirical evaluation results. The empirical evaluation is composed of two phases: (1) industry case study evaluation conducted in an IT organization, and (2) industry field survey with a cohort of experts in software engineering. The evaluation aims to determine the applicability and novelty of the DRA framework. The evaluation results indicate that the DRA is fit for its intended purpose and is expected to help both researchers and practitioners. The DRA uses can architect complex design models for the purpose of supporting the automated development stage of software applications. Software engineers can implement the DRA models for deploying complex IoT application to the multi-cloud environment in the current organization development parameters.

Keywords DevOps reference architecture · Multi-cloud IoT applications · DevOps framework · Design science research · Case study evaluation · Survey evaluation

Abbreviations

AAF	Average and above frequency
AAP	Average and above percentage
QEM	Qualitative evaluator matrix
QIM	Quantitative indicator matrix
QI	Quantitative indicator
CI	Continuous integration
CD	Continuous deployment
DRA	DevOps reference architecture
DSR	Design science research
IoT	Internet of things
MQTT	Message queuing telemetry transport
RSS	Really simple syndication
POC	Proof of concept

Introduction

DevOps has emerged as a practical approach to software development in the context of agile [1, 2]. DevOps enables automation, continuous integration, monitoring, and team collaboration [3, 4] to assist the fast deployment and delivery of quality software [5]. Internet of Things (IoT) is a digital technology that enables connecting a large number of physical devices or things as virtual objects over an established network [6]. There is an increasing interest in IoT applications within the overall context of a digital ecosystem that involves several heterogeneous devices and protocols [7]. There is also an increasing interest among organizations to adopt DevOps for IoT application deployment to multi-cloud [8, 9]. Multi-layer cloud and DevOps could prove useful for the IoT-applications that require frequent updates and real-time interactions with the IoT devices [10, 11]. Hence, the question is how to develop, deploy and govern IoT applications to multi-cloud within the overall parameters of the existing organization ecosystem? To address this complex problem, this research presents the DevOps reference architecture (DRA) [8].

✉ Georges Bou Ghantous
Georges.BouGhantous-1@uts.edu.au
Asif Qumer Gill
Asif.Gill@uts.edu.au

¹ University of Technology Sydney, Sydney, Australia

The DRA architectural design is founded on five models: (1) contextual, (2) conceptual, (3) logical, (4) physical, (5) operational. DRA was constructed using the guidelines of the design science research (DSR) method [12]. The adopted DSR [12] has six steps [8]; the paper explains the empirical evaluation (step 5) of DRA and discusses the DSR outcome (step 6) to determine the DRA contribution to both research and practice.

The main scope of this paper is limited to the DRA architecture design and empirical evaluation of the new DRA framework [8]. The empirical evaluation involves two main steps: (1) industry case study, (2) industry field survey. A case study template (CST) was developed, tested and then offered to the case study participant along with the instructions and guidelines for the testing of the DRA design. The survey was offered online to industry practitioners and experts from the software engineering community. The survey participants came from a cohort of international and local organizations.

There are two types of collected data in the empirical evaluation: (1) qualitative (case study and survey); (2) quantitative (survey). The qualitative data were examined to ascertain the correlation between DRA design models and a list of validation criteria. The validation criteria used in this research are explicitly applicable to the evaluation of DSR artefact outcomes [13, 14]. The quantitative data collected in the field survey were analyzed using the well-known statistical analysis techniques (frequency, percentage, and $\chi^2 p$ value). The quantitative results are plotted onto histogram tables that provide a visual presentation of the survey results.

This paper is organized as follows. First, the article outlines the research problem and discusses related work. Second, the research presents the DSR method. Third, the paper includes an overview of the updated DRA. Fourth, the study discusses the empirical evaluation results of the DRA. Finally, the document discusses the framework's applicability based on the evaluation results and concludes the article with future scope directions.

The research DRA empirical evaluation and the framework applicability are the main contributions of this research. The results of the empirical evaluation aim to deliver sufficient proof that the DRA address the research problem and provide an effective solution for researchers and practitioners to automate the deployment of IoT application to multi-cloud.

Literature Review and Related Work

DevOps aims to improve collaboration and communication between Development and Operations [3]. DevOps provides a set of practices [15] to enable the automation of application deployment for timely release and delivery [16, 17].

DevOps offers broader support for the deployment of applications to the cloud [5, 18, 19] using a wide range of tools that enable automation and continuous integration (CI) [20].

There is growing interest in organizations to adopt DevOps for IoT [6, 21]. IoT, supported by cloud computing [22], aims to achieve interoperability and fast data exchange [23, 24]. Cloud offers PaaS (Platform as a Service) as a virtual platform for IoT application developers. It also provides a back-end solution to manage the vast data streams of IoT application data using Infrastructure as a Service (IaaS) and Software as a Service (SaaS) [25]. DevOps and IoT complex contexts present opportunities and challenges to cloud-computing [26, 27].

The value of IoT for enterprises resides in the fast deployment of IoT applications by developers [28], and by the effective seamless integration with other systems such as the cloud [29]. Models such as SysADL are explicitly designed to preserve a system-oriented [30] using an ADL based on SysML architecture. The IoT applications seem to be data-driven [31, 32]. The performance of IoT applications is deduced by measuring the latencies of interactions using communication protocols (e.g. MQTT, RSS, SSH, Wi-Fi, and mobile [33]) and by handling the increasing amount of data [34, 35].

Cloud computing enables ubiquitous, on-demand shared resources (Cloud API, configurations, and services) [36] that assist with the IoT-applications deployment and auto-scaling [25, 37]. Cloud computing can provide advanced resources to assist with the design of conceptual models to improve the autoscaling trends of software applications [38, 39]. The interplay and cooperation between fog (edges: support IoT devices) and the core (cloud) can be characterized by the integration between code and devices [29, 40]. Multiple clouds or multi-cloud is the integration of multiple cloud services in a single heterogeneous architecture. Organizations, developers, and researchers can benefit from open-source cloud platforms because they encourage the use of the multi-cloud through broader user access, flexibility, availability, and high-level quality of service (QoS) to the same or different applications deployed to multi-cloud [41, 42]. However, cloud applications are often hardwired to their native cloud API [43].

The major obstacle for adopting a multi-cloud strategy is vendor lock-in [43, 44]. Vendor lock-in may occur in two cases: when a cloud from the multi-cloud cohort hosts the deployment configuration and when a cloud from the multi-cloud cohort hosts the database. Several studies and frameworks have introduced innovative ideas to achieve heterogeneous architecture for continuous deployment to the multi-cloud [45]. For instance, CYCLONE [46] is a software stack that focuses on the areas of application deployment, management, and security and user authentication on the multi-cloud. Another model, CloudMF [23],

is an object-oriented domain-specific model tailored for IoT applications. The deployment process to the multi-cloud can follow specific migration patterns [47], such as multi-cloud refactoring and multi-cloud rebinding. The dynamic Data-Driven Cloud and Edge Systems (D3CES) approach enables real-time IoT data collection and provides feedback that promotes effective decision-making to deploy IoT applications to the cloud [47, 48]. IoT can benefit from multi-cloud services and techniques that enable portability and interoperability [49].

The literature review and related work draw our attention to further research in the possible integration of DevOps, multi-cloud, and IoT applications [50, 51]. DevOps adoption for IoT application deployment to multi-cloud requires concrete architecture and guidelines [19, 52]. This need highlights the following research problem and challenges:

- Automated IoT applications deployment to multi-cloud.
- Manage connectivity between the IoT application and IoT sensors.
- Avoid vendor lock-in for IoT application deployment to the multi-cloud.

The new proposed DRA framework addresses the research problem. This paper presents the updated DRA version and its empirical evaluation.

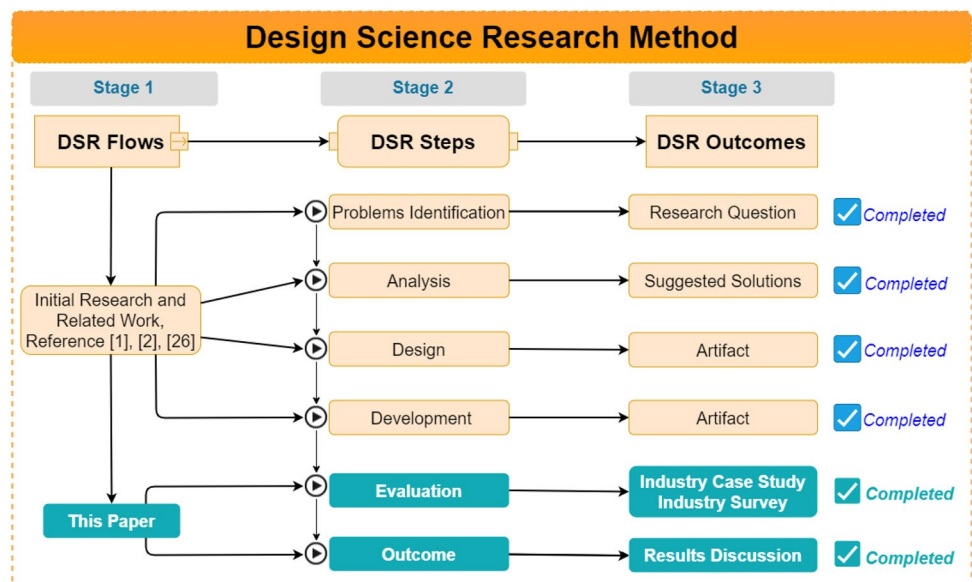
Research Method

This research adopts a well-known DSR method [12, 53], which is a system of principles, practices, and procedures pertained to a specific branch of knowledge to produce and

present high-quality research artefacts. The DSR aims to provide verifiable contributions through the design, development, and evaluation of the proposed DRA artefact. The artefact development may involve the review of existing theories and knowledge to develop a solution or artefact for the intended purpose and audiences. The DSR is composed of three primary stages (Fig. 1):

- Stage 1—DSR main flows. The DSR is composed of two flows:
 - Literature review and related work analysis in “Literature review and related work”.
 - Related work in publications [3, 8, 54].
- Stage 2—DSR process steps. The DSR process in this thesis is composed of six steps:
 - Problem identification: Initial research into the background and related work helped identify the research problem and gaps and its underlying objectives in “Literature review and related work”.
 - Analysis: The related work and background research provided rich information about DevOps, multi-cloud and IoT.
 - Design: A general design model is created for the new DRA founded on DevOps concepts, and cloud infrastructure and services.
 - Development: DRA architectural model is developed based on the DevOps concepts and cloud services. The architectural model is not fixed and can be applied to numerous instances in multiple contexts.
 - Evaluation: The DRA is evaluated using the DSR evaluation criteria [13] 14 (Table 2). The evaluation

Fig. 1 DSR method



involves an industry case study and field survey. The evaluation results and the updated DRA are the main contributions of this paper.

- Outcome: The updated DRA and its contribution to the SE body of knowledge.
- Stage 3—DSR outcomes. The DSR process in this thesis is composed of six outputs:
 - Research problem (“Literature review and related work”).
 - Objective solutions (“Literature review and related work”).
 - Design artefact (“The DRA overview”).
 - Development artefact (“The DRA overview”).
 - Empirical evaluation (“Industry case study”).
 - Discussion (see “Industry survey”).

DSR Evaluation Method

The DRA has been evaluated using an empirical evaluation approach. The empirical evaluation includes an industry case study and survey. The case consists of five steps: design,

prepare, collect, analyze, and report. The industry survey consists of five steps: plan, prepare, develop, deliver, and report. The empirical evaluation process is presented in Fig. 2 (based on [55] [56]), whereas evaluation criteria are shown in Table 2 (based on [13, 14]).

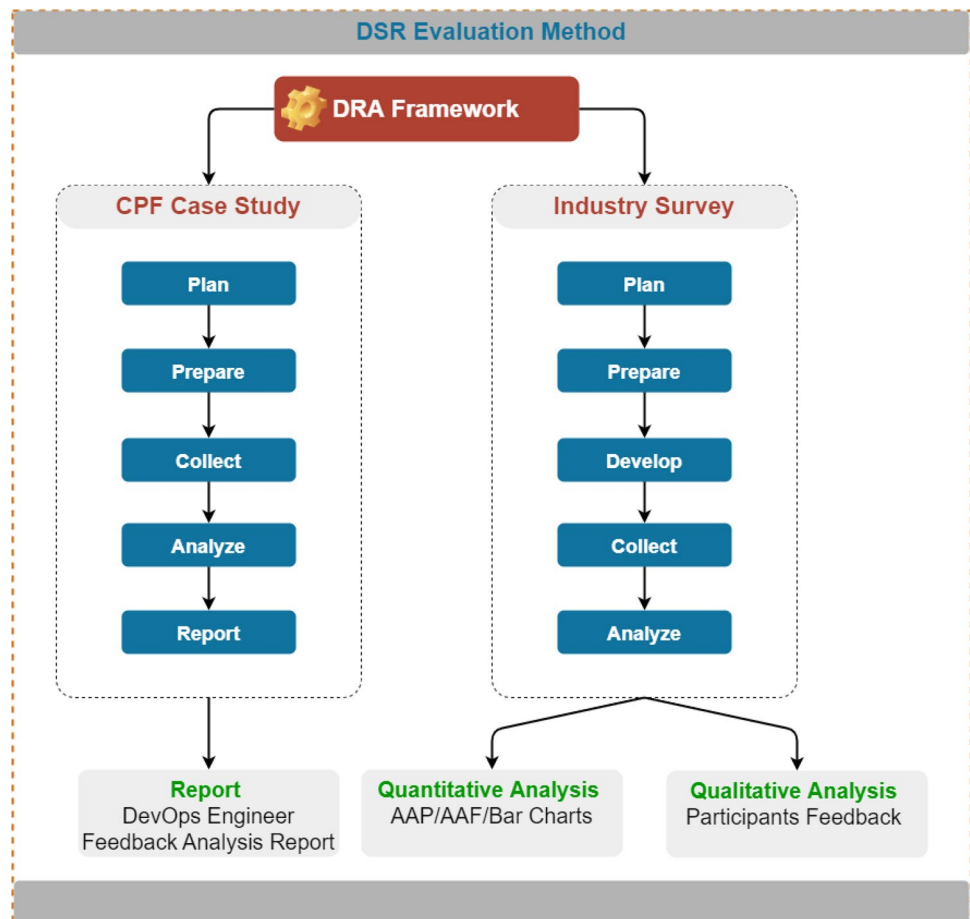
DRA empirical evaluation overview shows the evaluation steps (case study and survey). The evaluation data is analyzed to determine the applicability of the DRA.

Case Study Design

The case study approach is commonly used for software engineering research artefact evaluation [55], such as the DRA in this research. In software engineering, an artefact could be an architecture, method, or process model or software tool [56]. This research uses the interpretive case study approach [55], which includes the following steps.

1. Case study planning: Plan the case study and identify the objectives
2. Preparation for data collection: Prepare the data collection method used in the case study.

Fig. 2 DRA empirical evaluation



3. Collecting data: Present the case study implementation and data collection method.
 4. Data analysis: Analyze the qualitative data collected using CST and compare the participants' feedback to the evaluation criteria (Table 2).
 5. Reporting: The report summarizes the outcomes of the case study. The report table includes the testing steps of the case study and the description of each step. The report provides evidence of the DRA applicability in the real context of the case study organization.
2. Prepare: Identify the target participants (Ethical considerations are required)
 3. Develop the questionnaire: Industry survey questionnaires are defined by the researcher using artefact evaluation criteria [13].
 4. Collect: Present the survey collection method.
 5. Analyze the data: The survey evaluation is composed of two main steps: Survey Quantitative Evaluation and Survey Qualitative Evaluation.

Survey Design

The survey utilized in this research follows a commonly used survey structure [57]:

1. Plan: Outline the survey objectives (purpose, need, knowledge requirements).
1. Strongly agree: The participants consider the question claim very satisfactory.
2. Agree: The participants agreed with the statement.
3. Average: The participants somewhat agreed with the statement.
4. Disagree: The participants disagreed with the statement.
5. Strongly disagree: The participants strongly disagreed with the statement.

Table 1 Rating table

Qualitative rating	Quantitative rating
Strongly agree	5
Agree	4
Average	3
Disagree	2
Strongly disagree	1

Table 2 Evaluation criteria

Criteria	Description
Generalisations	DRA is general in the sense that it is not fixed to one situation or environment DRA can adapt to different situations and be used with varying stacks of technology DRA is instantiable and applicable to a class of problem situations
Usefulness	DRA is useful in an organization context DRA can be used as a blueprint for IT projects
Coverage	DRA provides sufficient explanation about DevOps adoption DRA provides setup and configuration guidelines for DRA instance pipelines DRA is a comprehensive design that provides the necessary practices for IT projects
Novelty	DRA offers new knowledgebase about DevOps adoption DRA offers CI-Broker to support multi-cloud deployments DRA improve agile application delivery using a DevOps approach
Relevance	DRA is relevant for deployment IoT applications on the multi-cloud at the industry level DRA framework new knowledge is relevant for teaching, industry and research DRA is relevant for organisations seeking to improve agility using a DevOps approach
Re-usable	DRA can be replicated and re-configured for a class of problem situations DRA design can be re-used to create instances for a class of situations DRA instance pipeline can be configured using a class of several tools
Importance	DRA enables end-to-end automation process and allows decentralised control DRA uses integrated DevOps and cloud tools to support IoT apps process DRA models are a high-level design that can be replicated for any software application

Statistical Analysis Method

This research used a statistical method to analyze the survey data. The statistical approach is better suited to bring out essential insights from the survey data. According to Hyndman [57], ‘Statistics is the study of making sense of data.’ The statistical formulas used to analyze the numerical survey data are explained in (1), (2), and (3).

Equation (1) describes the Chi-squared that calculates the probability p value ($0 \leq p \leq 1$) and compares it to a critical value $\alpha = 0.01$. If p value $< \alpha$, then H_0 is rejected and H_1 is accepted.

$$\text{Chi}^2 \text{ or } X^2 = \sum \frac{(O - E)^2}{E} \quad (O = \text{frequency and } E = \text{expected value}) (p\text{-value} < 0.01), \quad (1)$$

$E = \Sigma O/N$ (O = frequency and N = total number of observations).

The p value determines if the null hypothesis H_0 is accepted or rejected based on a critical value $\alpha = 0.01$.

If p value $< \alpha$, then H_0 is rejected and H_1 is accepted, and there is a positive association between the test variables (DRA models) and the evaluation criteria (see Table 2).

[If p value $< 0.000\epsilon$ (ϵ is a small number), then p is mathematically corrected to $p < 0.001$].

H_0 (null hypothesis): there is no association between the test variables and the evaluation criteria.

H_1 (alternative hypothesis): test variables and the evaluation criteria are positively associated.

Equation (2) describes the sum of frequencies of the participants’ scoring three and above in the rating table (Table 1).

$$\text{AAF} = \sum \text{Frequency (Ratings } \geq 3) \quad (2)$$

AAF is the sum of all participants responses [Average (3) + Agree (4) + Strongly Agree (5)]

Equation (3) describes the sum of percentages of the participants’ scoring three and above in the rating table (Table 1).

$$\text{AAP} = \sum \text{Percentage (ratings } \geq 3) \quad (3)$$

AAP is the sum of all percentages of responses [Average (3) + Agree (4) + Strongly Agree (5)]

Evaluation Criteria

The purpose of the empirical evaluation is to evaluate the DRA models. The evaluation method used in this research comprises of two phases:

- The survey questionnaire sets a relationship to the evaluation criteria.
- Case study participants’ feedbacks relationship to the evaluation criteria.

The evaluation criteria elements are selected based on models used in related work [13, 14]. The survey questionnaire sets are developed to assess the DRA models against the chosen criteria (Table 2). The feedback from the participants’ in the survey and the case study. The feedback was cross-examined to determine their relationships with the evaluation criteria (Table 2).

The DRA Overview

The DRA is founded on the concepts and practices of DevOps [2, 3]. DRA uses the multi-cloud ecosystem to support the deployment of IoT applications [21, 58]. DRA enables automated deployment of IoT-applications to multi-cloud [8]. The initial work-in-progress version of the DRA was published [8] to obtain early feedback from the research community before commencing this empirical study. This paper presents the updated version of the DRA with empirical evaluation results. The updated DRA architecture design is composed of five models [8]: contextual (model 1), conceptual (model 2), logical (model 3), physical (model 4), and operational (model 5) (Figs. 3, 4).

DRA Contextual Model

The DRA contextual (Fig. 3) model (model 1) describes the relationship between DevOps, Multi-Cloud, and IoT at a higher level. DRA contextual model is founded on

DevOps concepts [3]. Multi-Cloud offers DRA broader user access to virtual servers and a vast array of services [8]. DevOps approach and Multi-Cloud technologies aim to facilitate and support the IoT application deployment [5, 10, 18, 21].

Fig. 3 DRA contextual model

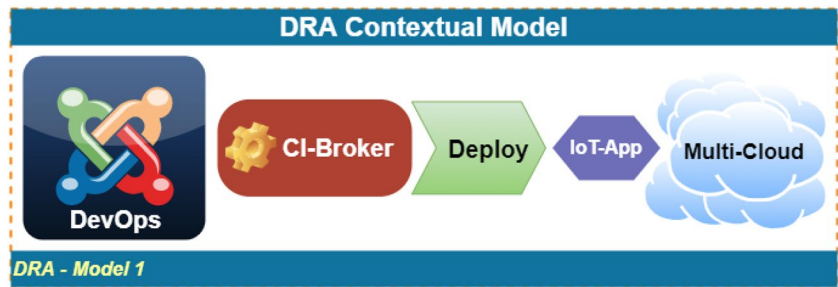


Fig. 4 DRAv2.0 Instance (Based on Bou Ghantous & Gill 2018)

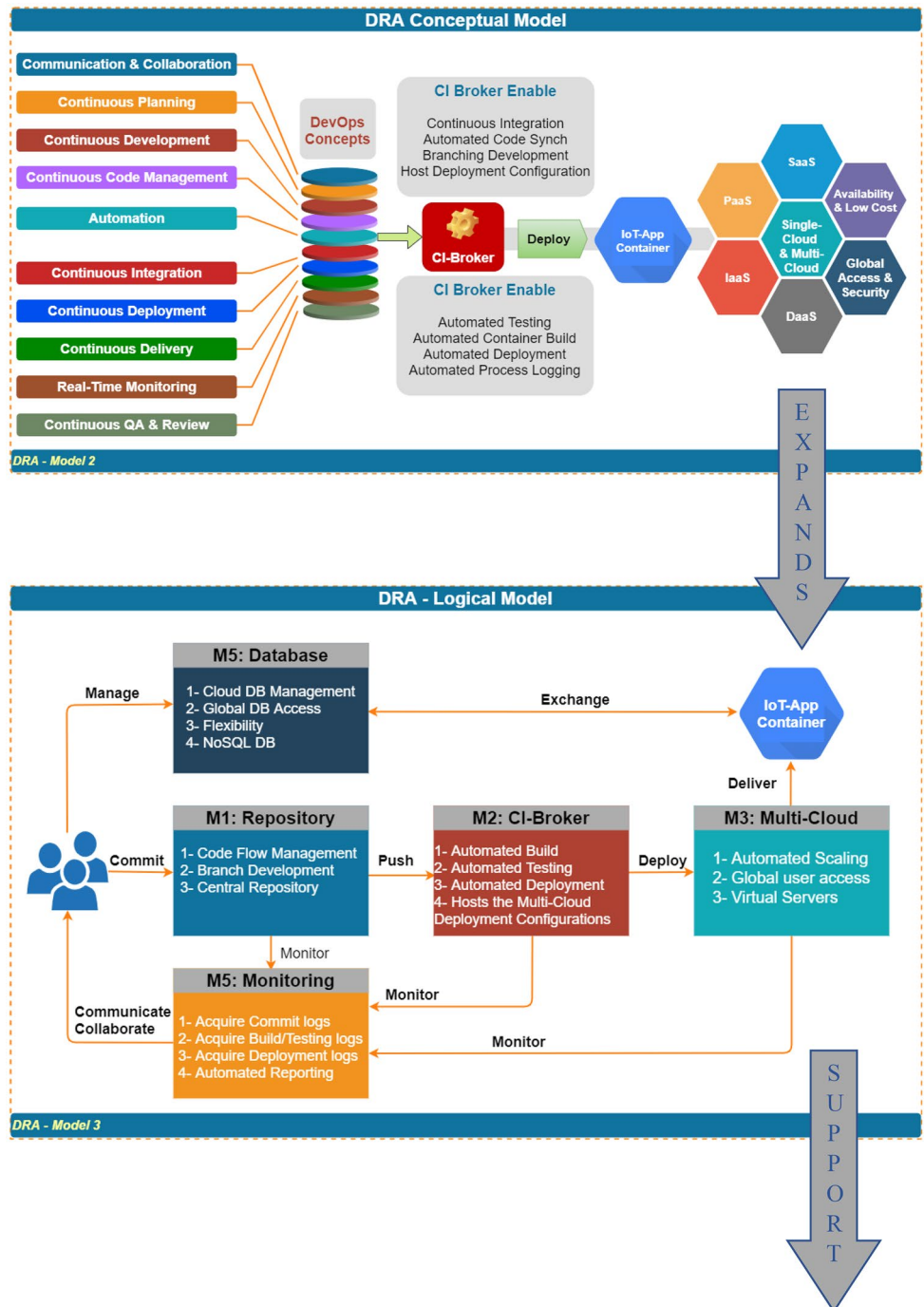
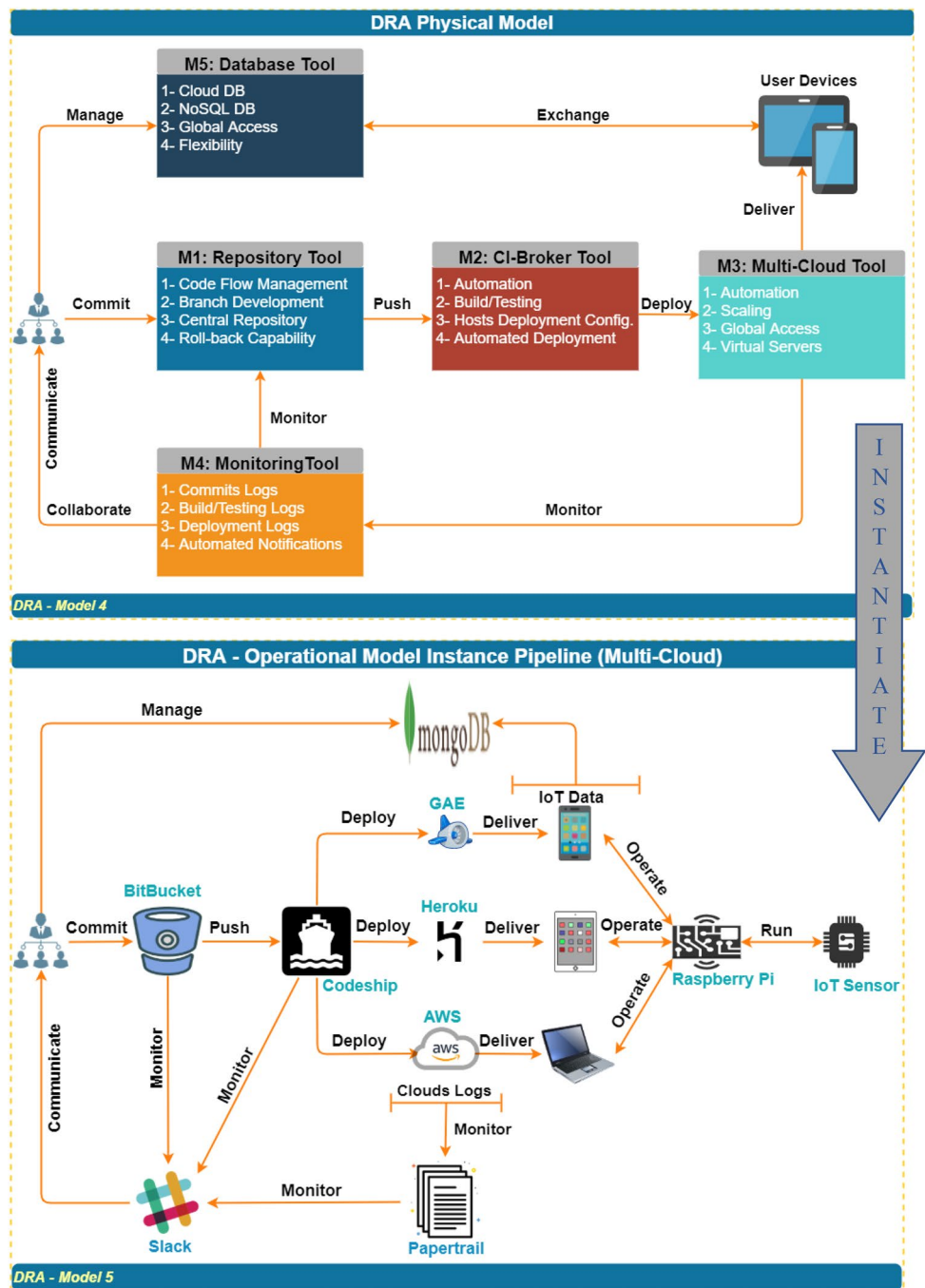


Fig. 4 (continued)



DRA Conceptual Model

The DRA Conceptual Model (model 2) (Fig. 4) expands in detail the high-level contextual model components [DevOps, Multi-Cloud, and IoT] [8]. The DRA context system enables:

- Automation of agile software development.
- Automation of application deployment process using CI-broker (continuous integration broker).
- Automated faster application delivery.

- Automated and integrated testing of the application.
- Enhance team experience and improve collaboration and communication.
- Enable real-time monitoring of the application.

The CI broker component was introduced in the DRA to address the research gap (“Literature review and related work”, e.g., vendor multi-cloud deployment, connectivity, and lock-in). With the research problem in mind, it was essential to devise an approach to deploy IoT applications

to the multi-cloud and avoid vendor lock-in, which occurs when a cloud vendor hosts the deployment configurations and when a cloud vendor hosts the database. The CI broker is an essential and novel part of the DRA conceptual model. For instance, the CI broker enables automation (build, testing, logging, deployment), CI, branching development, and automated code synchronization. Most importantly, it hosts the deployment configurations for the IoT application independent of any cloud technology and vendor. The CI-Broker also seems to address the issue of a single or fixed cloud DevOps environment and aims to enable the adaptability, integration, and interoperability of the DevOps approach for supporting the multi-cloud IoT application deployment. The CI broker packages the IoT application in a container and deploys it to the multi-cloud platforms. Thus, the DRA is generic, enabled by CI broker, to be able to be used for different contexts.

DRA Logical Model

The DRA logical model (model 3) further unpacks the conceptual model and is organized into components (M1–M5). The logical model architecture uses DevOps and cloud to create a functional model. The logical model illustration shows how DevOps practices [3] are transformed into features and functions to support the IoT application deployment to multi-cloud (Fig. 4).

DRA Physical Model

The DRA physical model (model 4) (Fig. 4) is an implementation of the DRA logical model. DRA Physical model creates a tangible design based on the Logical components (M1–M5). DRA physical model can be sub-divided into three tiers:

1. DevOps Team tier:
 - M1 enables team communication and real-time notification. M1 receives build/test logs from M2 and deployment and performance logs from M4.
2. Cloud tier—composed of:
 - M2 is the CI-Broker that handles the build, testing, and deployment of the application.
 - M3 is the deployment cloud(s) platform.
 - M4 is the monitoring and tracking platform.
 - M5 is the data management cloud. Applications exchange data NoSQL or SQL with M5. The database using the M5 model is managed separately from the deployment cloud to avoid vendor Lock-in.
3. User tier—represents the user devices operating the application and exchanging data with M5.

The DRA Physical Model can be instantiated to create an end-to-end deployment pipeline (Operational Model) using numerous technology stacks and DevOps tools.

DRA Operational Model

The operational model (model 5—live instance) of the DRA is based on the physical model. The DRA operational model pipeline instance (Fig. 4) is configured using an integrated set of DevOps tools [3, 8]. The DRA pipeline provides automated IoT-application deployment to multi-cloud [8].

The operational aspects (M1–M5 components) of the DRA pipeline are explained below:

1. The IoT-app code is pushed from M1 to M2.
2. M2 (CI-Broker) deploys the IoT-application to M3.
3. M3 is the deployment platform of DRA (multi-cloud).
4. M4 monitors the build, test, and deployment logs.
5. M5 store the IoT-application NoSQL data and provide central management.
6. M1 notifies the DevOps team with the logs from M4.

Figure 4 is composed of four images representing the DRA2.0 models (model 2 to model 5). Figure 4 is refactored into separate images to improve the resolution and view of the DRA design models (model 2 to model 5) relationships as follows:

Industry Case Study

The DRA evaluation was conducted using the case study template for case study organization CPF (coded name) (Link) (see “Appendix”). The data collected from the case study is qualitative feedback about the DRA. The case study process and results are explained below.

Plan

The case study was designed to demonstrate the capabilities of the DRA for the CPF organization. It includes the following:

- Identify the case study organization: The case study is conducted at the Australian organization [CODE_NAME: CPF] based in Sydney. Date: 24/04/2019
The organization name was kept anonymous—see ethics approval stored on CloudStor (Link).
- Case study organization context: CPF provides a single AWS cloud-based modelling platform for digital strategy, architecture, and project delivery. CPF aims to adopt a multi-cloud environment for its modelling platform to

address its customer needs. They also have a keen interest in emerging IoT applications.

- Need, and problem: Need a DevOps approach and middleware broker to deploy their platform features to a multi-cloud environment to meet their different customer needs and also to avoid cloud vendor lock-in.
- Solutions: DRA seems to address the need, as mentioned above, for the case study organization's need.
- Objective: The objective of the researcher is to demonstrate the applicability of the research-based DRA framework to the practical CPF's organizational context. The CPF case study organization aims to have a functional DevOps architecture and solution that support the deployment of their modelling platform to multi-cloud.
- DRA Proof of Concept (POC) demo and presentation: To demonstrate the applicability of the DRA framework, a proof of concept presentation and demo (including video of the demo) were developed in alignment with the case study need. The demo successfully demonstrated the deployment of a pre-developed sample IoT application to a multi-cloud environment. This demonstration shows the applicability of the DRA for the case study organization.
 - Presentation Slides (anonymized): [Link](#)
 - Demo Video YouTube video: [Link](#)

Prepare

The case study template (see CST Link) was prepared to collect the DRA evaluation data from the case study organization.

Collect

An evaluation session was organized with the CPF case study organization. They appointed the DevOps lead from their organization, which has expert knowledge about DevOps. The CPF DevOps lead participated in this evaluation and provided the relevant feedback data about the applicability of the DRA to their organization. The total duration of the data collection, including presentation and demo, was approximately 120 min. The case study data was stored in CloudStor, which is the recommended cloud storage (Link). The following sections discuss the evaluation feedback.

DRAv2.0 Architecture

The DRAv2.0 architecture was presented to the case study organization CPF. DRAv2.0 is composed of 4 design architecture models: conceptual, logical, physical, and operational. The expert from the CPF reviewed the design and provided positive feedback with further opportunities for

improvements. Feedback about the DRA architecture is provided below.

“It has been very well thought and process-driven. I think it would be excellent to include some controls which could be used in the case to re-deploy or even roll back to a previous version in an automated fashion”.

DRA Operational Model Pipeline

In this step, CST provides a checklist for the DRAv2.0 pipeline implementation. The participants may re-use the recommended tools-set or configure DRAv2.0 with other tools of choice (see CST Link). Feed-back about the pipeline is noted below.

“Tool used in Operation model pipeline are industry used tools and are an excellent choice for the DRA Operation Mode Pipeline.”

“Configuration template is easy to use and can be replicated.”

Software Component

DRAv2.0 can be configured to deploy software applications, including IoT applications, to a multi-cloud environment. The CST provides the participants with a demo application to test the DRAv2.0 architecture. The demo application source code can be accessed from the code repository using public access (see CST Link). The following feedback was received on the software components.

“Testing software component is functioning properly.”

Hardware Component

The CST provides information about the IoT-devices and network used to test the deployment of the IoT software application to multi-cloud. The IoT-network (see CST Link) is configured to provide proof of the concept of the DRA operation model and its applicability for deploying the IoT application to multi-cloud. However, one may use their own choice of IoT-devices and sensors to test the IoT application deployment using the DRAv2.0. The following feedback was received from the case study organization.

“Testing hardware component is functioning properly and responding appropriately.”

Overall Feedback

Based on the feedback, it can be suggested that DRA is fit for its intended purpose. Overall feedback about the DRA is noted below.

“Demo was easy to understand and well presented.”
 “DRA framework would help organizations understanding DevOps methodologies and agile application deployment and delivery.”

Analyze

The case study data was analyzed and presented in Table 3. The analysis method is a cross-examination comparison between the feedbacks and the case study evaluation criteria in Table 2. This analysis aims to connect or relate evaluation criteria to the organization’s feedback. The output of the analysis is organized into three columns: “Interpretation,” the researchers’ interpretation of the “participant feedback,” and “evaluation criteria.”

Report

The case study report aims to conclude the data about the DRA framework and its applicability. The report summary is presented in (Table 4).

Industry Survey

The survey (see “Appendix”) was used to evaluate the DRA from practitioners’ perspectives. Thus, data were collected from a broad audience [55] as opposed to a single organization case study. The survey was offered online to local and international industry experts. The survey was conducted using the following steps [57]:

- Plan survey
- Prepare the delivery method
- Develop the survey questionnaires
- Collect the data
- Analyze the data

Plan

The plan is to obtain experts’ feedbacks about the DRA framework models. The survey (Link) was offered to participants and experts from the IT industry specialized in the areas of software engineering, DevOps, Cloud computing and architecture, and IoT.

Prepare

The survey was offered online (Link) to industry experts who were contacted using the author’s LinkedIn account. The survey was opened between January 2019 and June 2019. A total of 82 participants came from different companies located in Australia and several other countries.

Table 3 Case study analysis

Participant feedback	Interpretation	DRA categories
‘It has been very well thought and process-driven. I think it would be excellent to include some controls which could be used in the case study to re-deploy or even roll back to a previous version in an automated fashion’	This feedback indicates that DRA design models are well thought of and could be improved with controls to enable automated rollback. DRAv2.0 enables automated deployment and automated rollback to stable software versions	Usefulness
‘Tools used in Operation model pipeline are industry used tools and are an excellent choice for the DRA Operation Mode Pipeline’	This feedback indicates that the DRA toolset choice is excellent for the instance pipeline	Usefulness
‘Configuration template is easy to use and can be replicated’	This feedback indicates that the DRA is easy to use in the organisation. It also suggests that the DRA is applicable and instantiable	Reusable
‘Testing the software component is functioning properly’	This comment indicates that the DRA IoT application is functioning as intended on the multi-cloud	Usefulness
‘Testing hardware component is functioning properly and responding appropriately’	This comment indicates that the IoT devices are interacting with IoT application deployed on the multi-cloud	Usefulness
‘Demo was easy to understand and well presented’	This feedback indicates that the organisation’s DE seems to be satisfied with the DRA demo package used for presenting the framework operations	Coverage
‘DRA framework would help organisations understanding DevOps methodologies and agile application deployment and delivery’	This feedback indicates that the DRA provides helpful new knowledge for organisations about the DevOps approach and agile application deployment and delivery process	Coverage usefulness

Table 4 Case study report

Case study 01	Description
Organisation	CPF [CODE-NAMED]
Test date	24/04/2019
Organisation context	CPF efficiently consolidates and integrates the digital strategy, solution designs and project delivery across the portfolio. CPF aims to provide a cloud-based modelling platform to enable the business transformation from strategy to execution
Test team (TT) (Participants)	DE at CPF who is involved in the company business product models
Organisation's need	CPF need DevOps approach to deploy their platform features to the multi-cloud environment for different customers
Test objective	The objective is to evaluate the applicability of the DRA in the practical organisational context. The organisation objective is to have a working DevOps environment based on a cloud platform that enables the automation of software deployment
Test case question	How can the application features be deployed to the multi-cloud using DevOps?
Test package (Pre-prepared)	To evaluate the DRA framework, a presentation slide pack and demo were developed to demonstrate the deployment of a predeveloped sample IoT application to the multi-cloud environment: Demo YouTube video: Link Presentation slides: Link
Main test component	The participant (DE) evaluated the following DRA components: <ol style="list-style-type: none"> 1. DRAv2.0 architecture 2. DRA operational model pipeline 3. Software components 4. Hardware components
Test method	Case study template (see " Appendix ") Industry survey (see " Appendix ")
Test duration	120 min (presentation, demo, survey, case study)
Data type	Qualitative feedback provided by the organisation's DE
Key activities	TT verifies that DRA supports DevOps concepts and practices TT verifies that DRAv2.0 toolset are reusable TT verifies that DRAv2.0 instance pipeline enables CI using CI broker TT verifies the automated deployment and delivery of the IoT-application TT verifies the IoT app interaction with IoT-devices TT verifies that DRA design models can be reusable and instantiable TT verifies that DRA conceptual model offers new knowledge for the organisation to help to understand the DevOps approach
Expected outcome	The expected outcome of the case study is to determine that the DRA provides possible solutions to CPF's needs and, by extension, verify that the framework addresses the research gaps
Actual outcome	The actual outcome is determined in the case study analysis (Table 3). The participant (DE) from CPF imparted valuable qualitative data (as feedback) about the DRA. The cross-examination between the feedback and Table 2 indicates that the evaluator (DE) considers DRA design models reusable in the organisation context and easy to configure. The participant suggested that the framework offers new knowledge about DevOps adoption. The result indicates that the framework is not fixed to a context and may be generalised as a conceptual design to fit the purpose of the organisation's context. The case study results indicate that the DRA framework seems to provide an adequate solution to deploy IoT applications on the multi-cloud using the DevOps approach

The demographic representation of the participants ([Link](#)) includes information about their professional experience, their organization location, and years of experience in their IT field.

Develop

The survey is composed of nine questionnaire-sets ([Link](#)) as follows:

- Q1-Set: DRA contextual model (5 questions)

- Q2-Set: DRA conceptual model (6 questions)
- Q3-Set: DRA logical model (5 questions)
- Q4-Set: DRA logical model features (9 questions).
- Q5-Set: DRA physical model (5 questions)
- Q6-Set: DRA operational model (8 questions)
- Q7-Set: DRA usefulness feedback (2 questions)
- Q8-Set: DRA suggested improvements (1 question)
- Q9-Set: DRA overall feedback (2 questions).

Collect

The survey (Link) generated two types of data:

- **Quantitative Data:** Represents the participants’ responses for each questionnaire-set (Q1-Set, Q2-Set, Q3-Set, Q4-Set, Q5-Set, Q6-Set, Q7-Set, and Q9-Set). The responses are transformed into numerical data using the rating table (see Table 1).
- **Qualitative data:** Represents the participants’ feedback collected in questionnaire-sets Q7-Set, Q8-Set, and Q9-Set.

The collected survey raw data were stored on CloudStor (Link).

Analyze

The collected data from the survey were analyzed in two phases:

- **Quantitative data analysis:** Presents a detailed analysis of the DRA design models based on the data collected from each questionnaire-set. The collected data are used to test if the DRA models meet the criteria in Table 2 (Coverage, Relevance, and Importance).
- **Qualitative data analysis:** Presents the analysis of the quantitative data collected from (Q7-Set and Q9-Set). To complement the quantitative analysis, qualitative feedback was collected and analyzed for productive results and insights, including ideas for further improvement (Q8-Set).

Quantitative Data Analysis

The collected data were analyzed using the statistical formulas (see Eqs. 1, 2, and 3) to test if the DRA design models (contextual, conceptual, logical, physical, and operational) meet the evaluation criteria positively (see Tables 2, 5). The quantitative data analysis is composed of six statistical cards (Labeled Card_[index], see Fig. 5). Each statistical card (see Fig. 5) includes the following items:

- Item 1 Organize the questionnaire-sets according to their relationship with the evaluation criteria into tables labelled QT_[index].
- Item 2 Collect and map the survey quantitative raw data into tables labelled RT_[Index].
- Item 3 Arrange the RT_[index] tables into category rating tables labelled CT_[Index] based on the questionnaire-sets group in the QT_[index] tables.
- Item 4 Plot the RT_[index] tables into bar graphs labelled RF_[index].
- Item 5 Calculate the statistical values AAP, AAF for all the RT_[index] tables, and the Goodness of Fit χ^2 for the CT_[index] tables. (see Eqs. 1, 2, and 3).
- Item 6 Generate result summary reports labelled RS_[index]. The results reports summarize the analysis of the DRA design models.

Contextual Model Analysis: Card [8] RSI: Result Summary

The contextual model evaluation results in Tables 6 and 7 can be interpreted as follows:

- AAF=370 out of total=410 responses indicate that most of the participants agree that the DRA contextual model positively meets with test criteria.
- AAP=90.25% indicates that there is a high percentage of participants who agree that the DRA contextual model positively meets with test criteria.
- Coverage $p = 0.001 < \alpha = 0.01$. This result means that H_0 is rejected, H_1 is accepted, and the DRA contextual model seems to cover industry contextual design needs.
- Relevance $p = 0.001 < \alpha = 0.01$. This result means that H_0 is rejected, H_1 is accepted, and the DRA contextual model seems relevant.
- Importance $p = 0.001 < \alpha = 0.01$. This result means that H_0 is rejected, H_1 is accepted, and the DRA contextual model seems important.

The statistical values indicate that the participants consider the DRA contextual model relevant and essential design and that it covers the industry needs. Figure 6 illustrates the frequency of participants’ responses to add further visual insight into the results (Table 8).

Table 5 DRA contextual model questions group

QT [8]	Description	Criteria
Q1	Does the contextual model provide the overall scope and purpose of using a DevOps approach for IoT app and multi-cloud at a high level?	Coverage
Q2	Do you think DevOps is appropriate for deploying IoT apps to the multi-cloud environment?	Coverage
Q3	Are the model elements (technologies) sufficient for the context?	Coverage
Q4	Is the contextual model relevant to the DRA framework?	Relevance
Q5	Are the contextual model elements critical to the DRA framework?	Importance

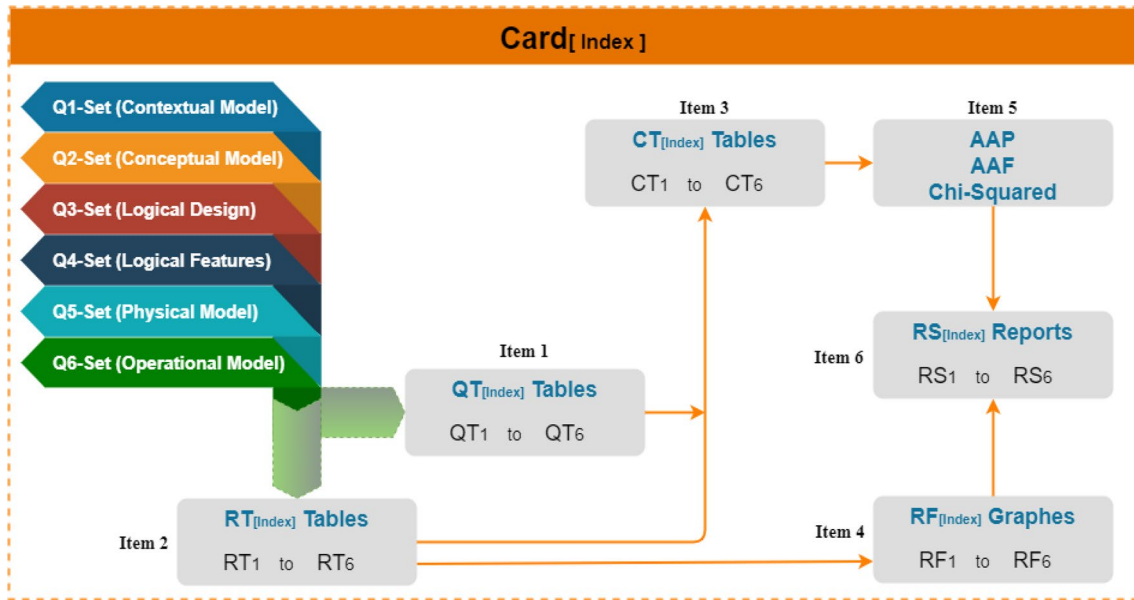


Fig. 5 Design models analysis cards

Table 6 Contextual questionnaire raw data

RT [8]	Q1	Q2	Q3	Q4	Q5	Row total	Percentage
Strongly disagree	3	1	5	3	3	15	3.66
Disagree	9	1	9	3	3	25	6.10
Average	14	3	17	15	15	64	15.61
Agree	40	23	32	44	42	181	44.15
Strongly agree	16	54	19	17	19	125	30.49
Column total	82	82	82	82	82	410	100.00

Table 7 Contextual group data

CT [8]	Coverage (Q1, Q2, Q3)		Relevance (Q4)		Importance (Q5)	
	O	E	O	E	O	E
Strongly disagree	9	49.2	3	16.4	3	16.4
Disagree	19	49.2	3	16.4	3	16.4
Average	34	49.2	15	16.4	15	16.4
Agree	95	49.2	44	16.4	42	16.4
Strongly agree	89	49.2	17	16.4	19	16.4
H ₀ is rejected for $p < 0.01$	$\chi^2 = 130.911$	$p < 0.00001$	$\chi^2 = 68.488$	$p < 0.00001$	$\chi^2 = 62.39$	$p < 0.00001$

Conceptual Model Analysis: Card [3] RS2: Result Summary Tables 9 and 10 present the statistical analysis results based on the participants’ responses. The conceptual model evaluation results can be interpreted as follows:

- AAF=382 out of total=410 responses indicate that most of the participants agree that the DRA conceptual model positively meets with test criteria.
- AAP=93.17% indicates that there is a high percentage of participants agree that the DRA conceptual model positively meets with test criteria.
- The p value for the test variables:
 - Coverage $p=0.001 < \alpha=0.01$. This result means that H_0 is rejected, H_1 is accepted, and the DRA conceptual model seems to cover industry design needs.

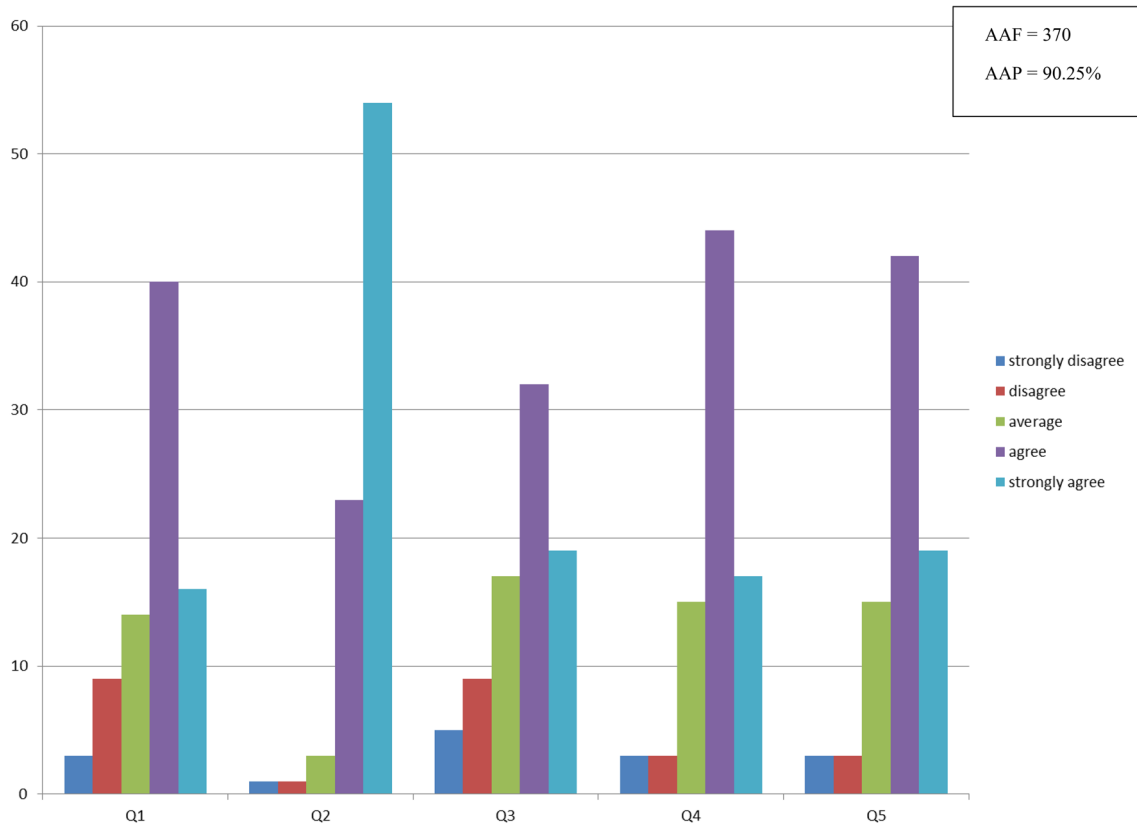


Fig. 6 Contextual data graph (RF1)

Table 8 DRA conceptual model questions group

QT2	Description	Criteria
Q1	Does the conceptual model provide enough components for DevOps?	Coverage
Q2	Does the conceptual model provide enough components for the cloud?	Coverage
Q3	Does the conceptual model provide enough components for the multi-cloud deployment platform of IoT apps?	Coverage
Q4	Is the conceptual model relevant for DRA framework?	Relevance
Q5	Is the conceptual model important for DRA framework?	Importance
Q6	Is CI-Broker a vital component for deploying IoT apps on multi-cloud?	Importance

Table 9 Conceptual questionnaire data

RT2	Q1	Q2	Q3	Q4	Q5	Q6	Row total	Percentage
Strongly disagree	1	1	2	1	1	2	7	1.71
Disagree	4	5	9	2	2	3	21	5.12
Average	15	14	17	13	12	11	67	16.34
Agree	39	44	40	46	42	31	203	49.51
Strongly agree	23	18	14	20	25	35	112	27.32
Column total	82	82	82	82	82	82	410	100.00

– Relevance $p=0.001 < \alpha=0.01$. This result means that H_0 is rejected, H_1 is accepted, and the DRA conceptual model seems relevant to the industry.

– Importance $p=0.001 < \alpha=0.01$. This result means that H_0 is rejected, H_1 is accepted, and the DRA conceptual model seems important for the industry.

Table 10 Conceptual group data

CT2	Coverage (Q1, Q2, Q3)		Relevance (Q4)		Importance (Q5, Q6)	
	<i>O</i>	<i>E</i>	<i>O</i>	<i>E</i>	<i>O</i>	<i>E</i>
$N=5; E=\Sigma O/N$						
Strongly disagree	4	49.2	1	16.4	3	16.4
Disagree	18	49.2	2	16.4	5	16.4
Average	46	49.2	13	16.4	23	16.4
Agree	123	49.2	46	16.4	73	16.4
Strongly agree	55	49.2	20	16.4	60	16.4
H_0 is rejected for $p < 0.01$	$\chi^2 = 172.902$	$p < 0.00001$	$\chi^2 = 82.024$	$p < 0.00001$	$\chi^2 = 125.39$	$p < 0.00001$

The statistical values indicate that the participants consider the DRA conceptual model relevant and essential design and that it covers the industry needs. Figure 7 illustrates the frequency of participants' responses to add further visual insight into the results (Table 11).

Logical Model Design Analysis: Card [13] RS3: Result Summary

- Tables 12 and 13 present the statistical analysis results based on participants' responses. The logical model design evaluation results can be interpreted as follows:
- AAF=386 out of total=410 responses indicate that most participants agree that the DRA logical model design positively meets with test criteria.
- AAP=94.14% indicates that there is a high percentage of participants agree that the DRA logical model design positively meets with test criteria.
- The p value for the test variables:

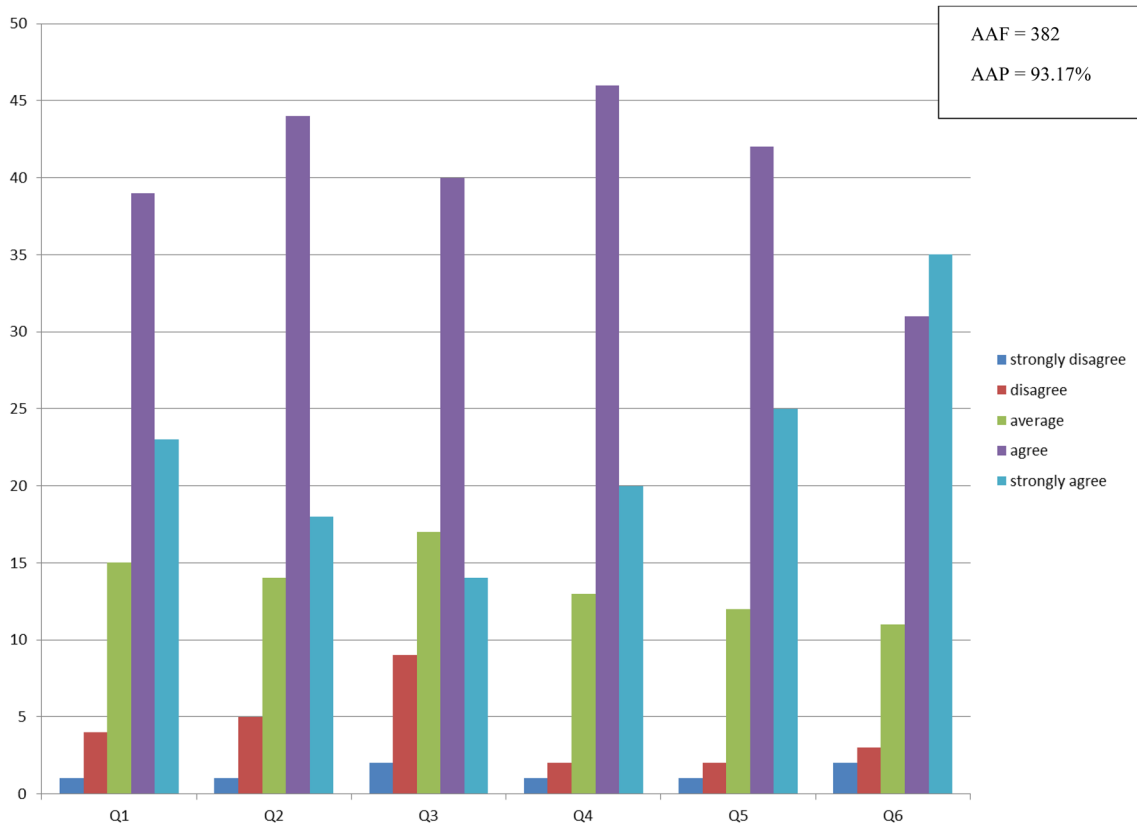


Fig. 7 Conceptual data graph (RF2)

Table 11 DRA logical model design questions group

QT3	Description	Criteria
Q1	Does the logical model provide enough components for DevOps?	Coverage
Q2	Does the logical model provide enough components for IoT apps deployment?	Coverage
Q3	Does the logical model provide enough components for the cloud platform?	Coverage
Q4	Is the logical model relevant to the DRA framework?	Relevance
Q5	Is the logical model important to the DRA framework?	Importance

Table 12 Logical design questionnaire data

RT3	Q1	Q2	Q3	Q4	Q5	Row total	Percentage
Strongly disagree	2	2	3	1	1	9	2.20
Disagree	3	4	5	1	2	15	3.66
Average	8	14	17	5	7	51	12.44
Agree	48	44	40	52	45	229	55.85
Strongly agree	21	18	17	23	27	106	25.85
Column total	82	82	82	82	82	410	100.00

Table 13 Logical design group data

CT3	Coverage (Q1, Q2, Q3)		Relevance (Q4)		Importance (Q5)	
	O	E	O	E	O	E
Strongly disagree	7	49.2	1	16.4	1	16.4
Disagree	12	49.2	1	16.4	2	16.4
Average	39	49.2	5	16.4	7	16.4
Agree	132	49.2	52	16.4	45	16.4
Strongly agree	56	49.2	23	16.4	27	16.4
H ₀ is rejected for $p < 0.01$	$\chi^2 = 206.724$	$p < 0.00001$	$\chi^2 = 116.78$	$p < 0.00001$	$\chi^2 = 89.22$	$p < 0.00001$

- Coverage $p = 0.001 < \alpha = 0.01$. This result means that H₀ is rejected, H₁ is accepted, and DRA logical model design seems to cover industry needs.
- Relevance $p = 0.001 < \alpha = 0.01$. This result means that H₀ is rejected, H₁ is accepted, and the DRA logical model design seems relevant to the industry.
- Importance $p = 0.001 < \alpha = 0.01$. Hence, H₀ is rejected, H₁ is accepted, and the DRA logical model design seems important for the industry.
- AAF=711 out of total =738 responses indicate that most of the participants agree that the DRA logical model features positively meet with test criteria.
- AAP=96.33% indicates that there is a high percentage of participants agree that the DRA logical model features positively meets with test criteria.
- The p value for the test variables:
 - Coverage $p = 0.001 < \alpha = 0.01$. This result means that H₀ is rejected, H₁ is accepted, and DRA logical model features seem to cover industry contextual design needs.
 - Relevance $p = 0.001 < \alpha = 0.01$. This result means that H₀ is rejected, H₁ is accepted, and the DRA logical model features seem relevant to the industry.
 - Importance $p = 0.001 < \alpha = 0.01$. Hence, H₀ is rejected, H₁ is accepted, and the DRA logical model features seem important for the industry.

The statistical values indicate that the participants consider the DRA logical model design a relevant and essential design and that it covers the industry needs. Figure 8 illustrates the frequency of participants’ responses to add further visual insight into the results (Table 14).

Logical Model Features Analysis: Card [59] RS4: Result Summary The numerical data in Tables 15 and 16 produced fundamental statistical values based on participants’ responses. The evaluation results can be interpreted as follows:

The statistical values indicate that the participants consider the DRA logical model features relevant and essential design and that it covers the industry needs. Figure 9

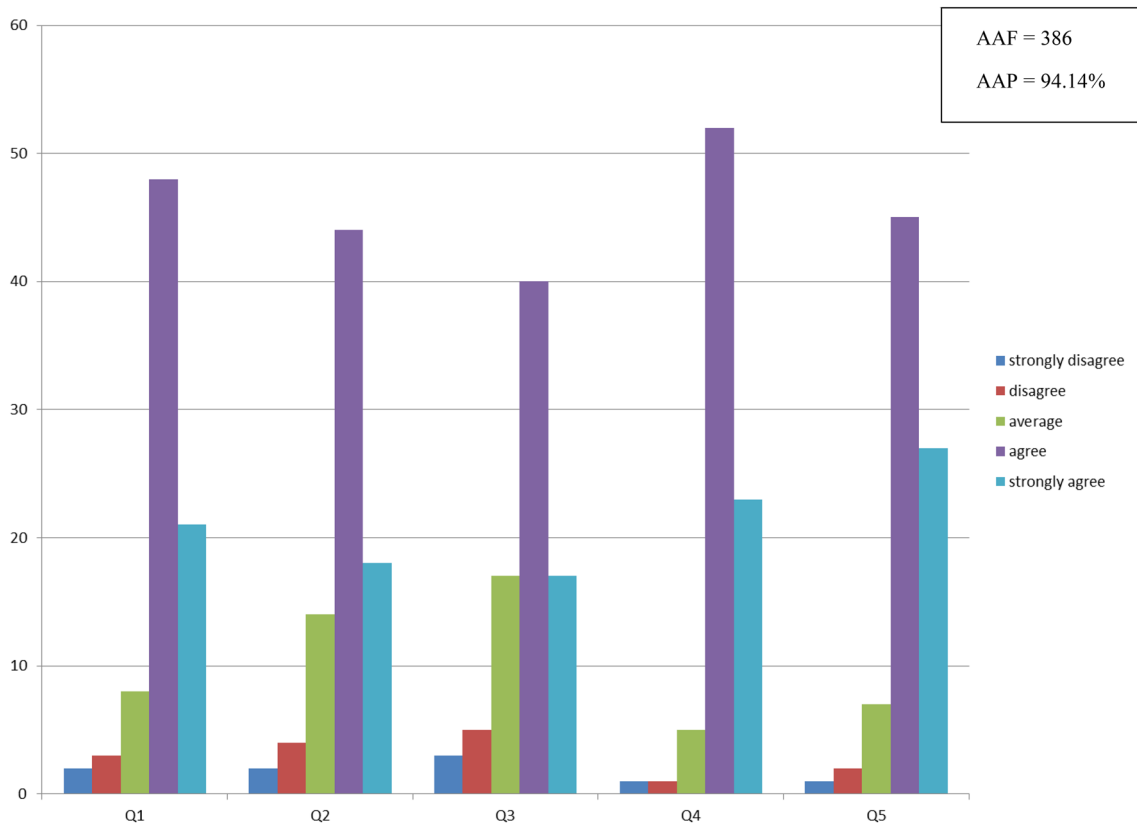


Fig. 8 Logical design data graph (RF3)

Table 14 DRA logical model functions questions group

QT4	Description	Criteria
Q1	DRA M1 automate code synchronisation for DevOps team	Coverage
Q2	DRA M2 enable automation for: repository update, build, testing	Coverage
Q3	DRA M2 enables deployment to M3 (using CI broker)	Coverage
Q4	DRA M3 automate scaling and application staging for users	Coverage
Q5	DRA M4 enable automated log capture from build, testing and deployment of IoT app	Coverage
Q6	DRA M5 provides cloud database management for DevOps team	Coverage
Q7	Do you think that the M1–M5 sub-models provide enough functions for the DRA framework?	Coverage
Q8	Do you think that the M1–M5 sub-models are relevant for the DRA framework?	Relevance
Q9	Do you think that the M1–M5 sub-models are essential for the DRA framework?	Importance

Table 15 Logical features questionnaires data

RT4	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Row total	Percentage
Strongly disagree	0	0	0	0	1	1	2	1	1	6	0.81
Disagree	2	3	3	3	3	2	1	2	2	21	2.85
Average	8	6	7	12	8	12	11	5	6	75	10.16
Agree	45	39	43	43	39	39	45	46	38	377	51.08
Strongly agree	27	34	29	24	31	28	23	28	35	259	35.09
Column total	82	82	82	82	82	82	82	82	82	738	100.00

illustrates the frequency of participants’ responses to add further visual insight into the results (Table 17).

Physical Model Analysis: Card [10] RS5: Result Summary
The numerical data in Tables 18 and 19 produced funda-

Table 16 Logical features group data

CT4	Coverage (Q6, Q7)		Relevance (Q8)		Importance (Q1–Q5, Q9)	
	<i>O</i>	<i>E</i>	<i>O</i>	<i>E</i>	<i>O</i>	<i>E</i>
<i>N</i> = 5; <i>E</i> = $\Sigma O/N$						
Strongly disagree	3	32.8	1	16.4	3	114.8
Disagree	3	32.8	2	16.4	18	114.8
Average	23	32.8	5	16.4	59	114.8
Agree	84	32.8	46	16.4	286	114.8
Strongly agree	51	32.8	28	16.4	208	114.8
H_0 is rejected for $p < 0.01$	$\chi^2 = 147.098$	$p < 0.00001$	$\chi^2 = 96.659$	$p < 0.00001$	$\chi^2 = 547.596$	$p < 0.00001$

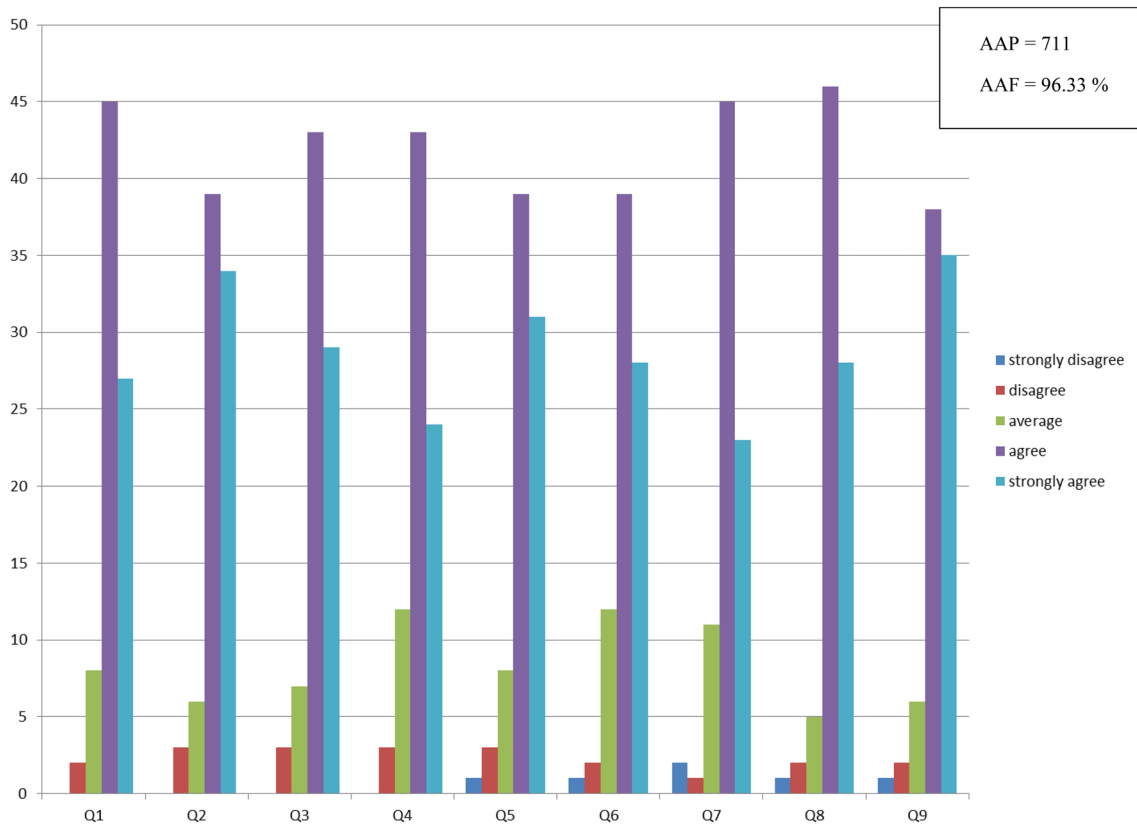


Fig. 9 Logical features data graph (RF4)

Table 17 DRA physical model questions group

QT5	Description	Criteria
Q1	Does the physical model provide enough features for DevOps?	Coverage
Q2	Does the physical model provide enough features for the cloud?	Coverage
Q3	Does the physical model provide enough features for IoT apps deployment?	Coverage
Q4	Is the physical model relevant to the DRA framework?	Relevance
Q5	Is the physical model important for the DRA framework?	Importance

Table 18 Physical model questionnaires data

RT5	Q1	Q2	Q3	Q4	Q5	Row total	Percentage
Strongly disagree	3	2	3	1	3	12	2.93
Disagree	2	4	7	4	1	18	4.39
Average	9	9	8	10	11	47	11.46
Agree	42	41	39	42	38	202	49.27
Strongly agree	26	26	25	25	29	131	31.95
Column total	82	82	82	82	82	410	100.00

Table 19 Physical model group data

CT5	Coverage (Q1, Q2, Q3)		Relevance (Q4)		Importance (Q5)	
	<i>O</i>	<i>E</i>	<i>O</i>	<i>E</i>	<i>O</i>	<i>E</i>
$N=5; E=\Sigma O/N$						
Strongly disagree	8	49.2	1	16.4	3	16.4
Disagree	13	49.2	4	16.4	1	16.4
Average	26	49.2	10	16.4	11	16.4
Agree	122	49.2	42	16.4	38	16.4
Strongly agree	77	49.2	25	16.4	29	16.4
H_0 is rejected for $p < 0.01$	$\chi^2 = 195.504$	$p < 0.00001$	$\chi^2 = 70.805$	$p < 0.00001$	$\chi^2 = 65.317$	$p < 0.00001$

mental statistical values based on participants' responses. The evaluation results can be interpreted as follows:

- AAF = 625 out of total = 656 responses indicate that most of the participants agree that the DRA physical model positively meets with test criteria.
- AAP = 95.28% indicates that there is a high percentage of participants agree that the DRA physical model positively meets with test criteria.
- The p value for the test variables:
 - Coverage $p = 0.001 < \alpha = 0.01$. This result means that H_0 is rejected, H_1 is accepted, and the DRA operational model seems to cover industry design needs.
 - Relevance $p = 0.001 < \alpha = 0.01$. This result means that H_0 is rejected, H_1 is accepted, and the DRA operational model seems relevant to the industry.
 - Importance $p = 0.001 < \alpha = 0.01$. Hence, H_0 is rejected, H_1 is accepted, and the DRA operational model seems important for the industry.

The statistical values indicate that the participants consider the DRA logical model features relevant and essential design and that it covers the industry needs. Figure 10 illustrates the frequency of participants' responses to add further visual insight into the results (Table 20).

Operational Model Analysis: Card [6] RS6: Result Summary The numerical data in Tables 21 and 22 produced

fundamental statistical values based on participants' responses. The evaluation results can be interpreted as follows:

- AAF = 625 out of total = 656 responses indicate that most of the participants agree that the DRA operational model positively meets with test criteria.
- AAP = 95.28% indicates that there is a high percentage of participants agree that the DRA operational model positively meets with test criteria.
- The p value for the test variables:
 - Coverage $p = 0.001 < \alpha = 0.01$. This result means that H_0 is rejected, H_1 is accepted, and the DRA operational model seems to cover industry contextual design needs.
 - Relevance $p =$ at $0.001 < \alpha = 0.01$. This result means that H_0 is rejected, H_1 is accepted, and the DRA operational model seems relevant to the industry.
 - Importance $p = 0.001 < \alpha = 0.01$. Hence, H_0 is rejected, H_1 is accepted, and the DRA operational model seems important for the industry.

The statistical values indicate that the participants consider the DRA operational model relevant and essential design and that it covers the industry needs. Figure 11 illustrates the frequency of participants' responses to add further visual insight into the results.

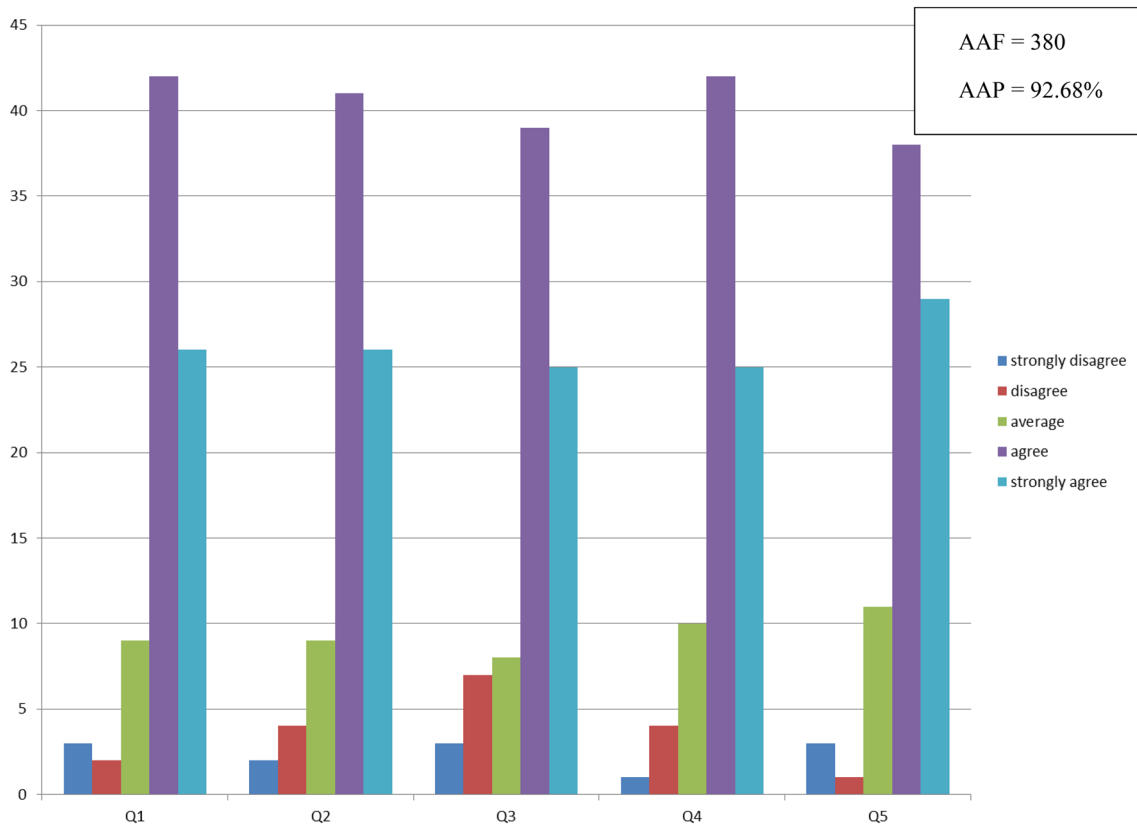


Fig. 10 Physical model data graph (RF5)

Table 20 DRA operational model questions group

QT6	Description	Criteria
Q1	Does the pipeline provide enough components to support DevOps?	Coverage
Q2	Does the pipeline provide enough components to support multi-cloud deployment?	Coverage
Q3	Does the pipeline provide enough components to enable IoT app deployment on multi-cloud?	Coverage
Q4	Does DRA pipeline enable automated IoT app deployment on multi-cloud using Codeship as CI broker?	Importance
Q5	Is DRA pipeline tools integration relevant for the framework?	Relevance
Q6	Are the DevOps tools in the pipeline sufficient for the framework?	Importance
Q7	Does the DRA pipeline reflect the conceptual design model?	Relevance
Q8	Does the DRA pipeline provide all the functions and features defined in the Logical model?	Importance

Table 21 Operational model questionnaires data

RT6	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Row total	Percentage
Strongly disagree	2	3	3	2	2	1	2	2	17	2.59
Disagree	0	1	2	1	2	3	2	3	14	2.13
Average	11	11	12	9	10	13	10	10	86	13.11
Agree	37	38	39	42	38	41	34	38	307	46.80
Strongly agree	32	29	26	28	30	24	34	29	232	35.37
Column total	82	82	82	82	82	82	82	82	656	100.00

Table 22 Operational model group data

CT6	Coverage (Q1, Q2, Q3, Q6)		Relevance (Q5, Q7)		Importance (Q4, Q8)	
	<i>O</i>	<i>E</i>	<i>O</i>	<i>E</i>	<i>O</i>	<i>E</i>
$N=5; E=\Sigma O/N$						
Strongly disagree	9	65.6	4	32.8	4	32.8
Disagree	6	65.6	4	32.8	4	32.8
Average	47	65.6	20	32.8	19	32.8
Agree	155	65.6	72	32.8	80	32.8
Strongly agree	111	65.6	64	32.8	57	32.8
H_0 is rejected for $p < 0.01$	$\chi^2 = 261.512$	$p < 0.00001$	$\chi^2 = 132.098$	$p < 0.00001$	$\chi^2 = 142.159$	$p < 0.00001$

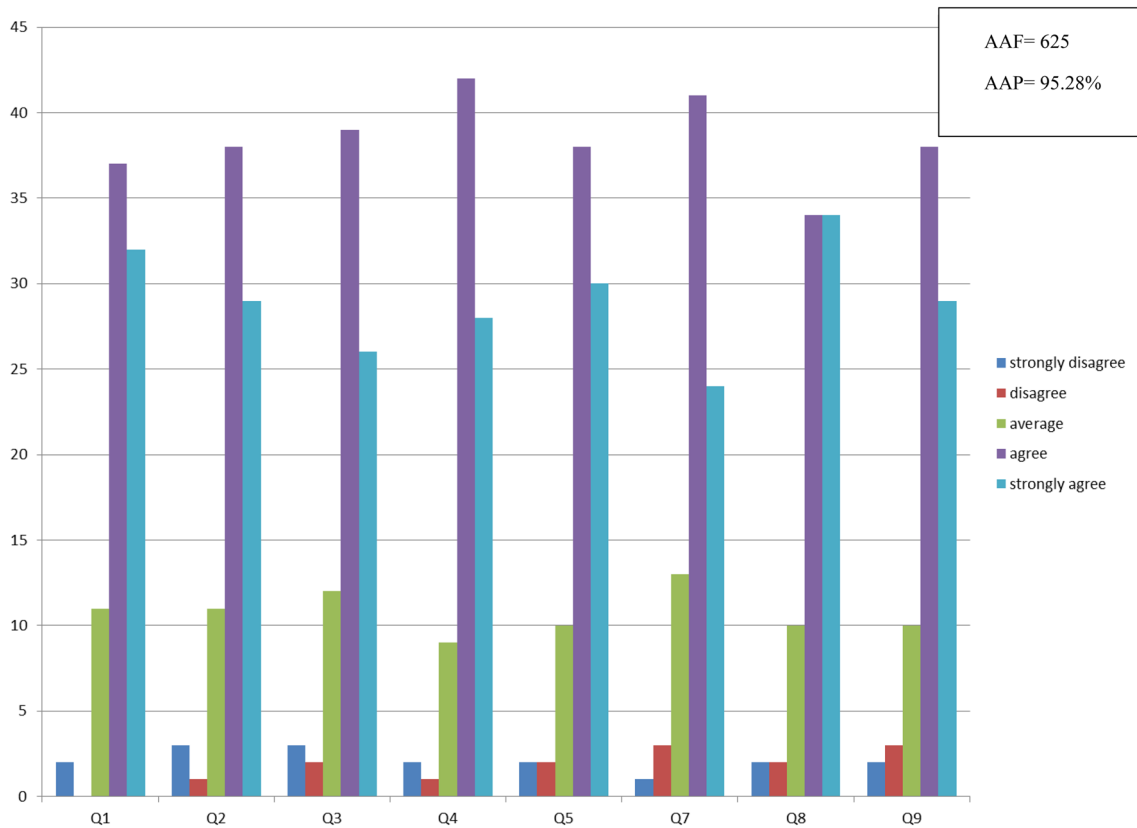


Fig. 11 Operational model data graph (RF6)

Qualitative Data Analysis

The qualitative data analysis is composed of two sections:

- DRA usefulness (for teaching, research, and industry) evaluation [Q7-Set]
- DRA overall feedbacks and ratings [Q9-Set]

DRA suggested improvements [Q8-Set] are used to indicate the future scope based on the participants’ feedback.

DRA Usefulness Feedback and Rating This section presents the analysis of the participants’ responses about DRA from a usefulness perspective in the industry, teaching, and research. The evaluation process is as follows:

- Collect and organize the feedback about DRA usefulness into Table 23.
- Analyze Table 23 feedback based on the occurrence of criteria elements (see Table 2) in the text using a cross-examination method.

Table 23 Q7-Set DRA usefulness feedback analysis

Criteria	Feedbacks: participants quotes	Interpretation
Importance Generalisations	One particular aspect that I think is very important in the DRA framework is the flexibility to choose the instantiations of each component (or indeed have several instantiations). Given the heterogeneous nature of the cloud and IoT environments, I think this is a critical feature	Flexibility Instantiations of each component Heterogeneous nature Cloud and IoT relationship important in the DRA
Usefulness Relevance Coverage	The models provided are all very useful as they deconstruct and disambiguate what is required in order to deploy any code as a single or multi-cloud application—this is achieved through presenting both abstract and concrete examples and clearly defining steps involved at each stage of the process	Useful models Disambiguate Deploy any code Adaptive to technology Process What is required
Usefulness Importance	The continuous integration and automated deployment to multi-cloud is very useful	Continuous integration Useful Automated deployment
Coverage Generalisations	All of it, it provides a comprehensive overview of what's needed for an IoT application deployment and management	What is needed for IoT-applications Comprehensive overview Enable IoT-applications deployment
Relevance Importance	Decentralized logging, cloud hosted CI, deployment to multiple hosting vendors, etc. are all acceptable modern solutions for the demonstrated problem	Decentralized logging Cloud CI Deployment to multiple vendors Acceptable modern solutions
Usefulness	DRA logical model specifications and DRA pipeline instance are very useful in the enterprise world, as they focus on the low-level implementation scheme	DRA logical, pipeline instance is very useful Low-level implementation
Usefulness	Video was good with the Raspberry Pi implementation, although implementing a "real" code change e.g. changing the frequency of flickering LEDs and showing the before and after results of the change would have been more concrete rather than just changing a text file, which we cannot verify has been deployed to the target destination in the video	Acceptable demo
Relevance	The fact that the entire deployment process is automated and seamless is really nice	Automated deployment process Seamless deployment process
Relevance	Interesting approach to combine DevOps and IoT	DevOps and IoT combination IoT interaction
Coverage Usefulness Generalisations	I do like this model for deployment of applications. I think it is quite extensive and applicable to developers, (ops) and IT management including project management	Extensive model Applicable IT and project management
Usefulness	It provides a framework that I'm pretty sure would be invaluable for people that don't know all the components and would like to implement it	Invaluable framework Easy to implement
Usefulness Relevance	An IDE integrated plugin which can deploy code readily to an IOT device, without being able to manage the entire infrastructure pipeline in between is a great value addition for any start-up working on IOT	IoT code deployment A valuable addition to IoT
Coverage Importance	Correctly identifies the benefits of DevOps Very appropriate tools and real use case implementation	DevOps benefit Appropriate tools Use case implementation
Relevance	It gives a clear idea of a path to production that could be used in IoT development process	Path to production IoT development process
Usefulness	Useful for IoT cloud solutions	Useful for Cloud-IoT
Importance Usefulness	The DevOps section is very good as all the tools are perfectly used as it will make good automation in architecture	Correct use of tools Architecture Useful Good DevOps section
Generalisations Relevance Usefulness	Having a high-level view of specifications, logic, modelling, and behaviour of the whole system is paramount to a solid implementation of the pipeline between IoT and Multi-Cloud	High-level view, logic, models IoT and Multi-cloud IoT interaction
Relevance Importance	It's good that you're creating a platform on the multi-cloud level. The idea of Automating end to end pipeline is a good thing	Multi-cloud platform End-to-end automation

Table 23 (continued)

Criteria	Feedbacks: participants quotes	Interpretation
Usefulness	I am really impressed with DRA I think the IoT environment are missing some DevOps structure like that provided by DRA, so it is very useful to guide teams to have a simple workflow to implement DevOps in IoT projects	Significance of DRA Useful DevOps implementation IoT projects
Relevance Usefulness	The value proposition that is to enable IoT architecture by the DevOps efforts in order to start using this amazing technology for an application like smart cities or even the set a cloud architecture for a smart house system	DevOps for IoT Cloud architecture IoT interaction May be used for the smart house system
Importance Usefulness	Definitely automation. When IoT makes to the top, it will be so hard to test things manually, and those things are going to be interacting directly and physically with users... So the automation process it's going to be a very good friend for the UX of anything related IoT	Automation Interaction It's going to be a very good friend for the UX of anything related IoT
Coverage	My feedback would be the same for DRA as well as for the framework, this info and the project itself is excellent, but has a lot of info, a lot of information very wide. Would also be good to watch more real-life scenarios	Framework Information is excellent
Usefulness Relevance Importance	This can be a useful reference document for IoT based Apps deployment with DevOps culture in the team process and toolset	Useful IoT-apps deployment DevOps culture
Generalisations	I thought it was a good overview of how DevOps can be applied with IoT	DevOps applied with IoT Overview
Importance	It is a good high-level design for the IoT cloud app deployment workflow	High-level design IoT cloud app deployment IoT interaction
Relevance Importance	Agility, CI/CD, automation, Speed	Agility, speed CI/CD Automation
Importance Relevance	Allowing users to interact with IoT devices remotely using cloud services. The ability to track usage through the use of paper-trail and automation of IoT device by using the cloud	IoT interaction Cloud services Cloud-IoT monitoring Useful Automation of IoT
Relevance Importance Coverage	Perfect pick up as industry is facing these issues as to how to integrate or implement DevOps transformation when it comes to Cloud and IoT related Apps. In a nutshell its part of digital transformation which is one of future in industry	DevOps transformation Cloud and IoT relationship IoT interaction Digital transformation

Table 24 DRA usefulness ratings

RT7	Research		Teaching		Industry		AAF	AAP
	<i>O</i>	<i>E</i>	<i>O</i>	<i>E</i>	<i>O</i>	<i>E</i>		
$N=5; E=\Sigma O/N$							Rows total	Percentage
Strongly disagree	1	16.4	2	16.4	2	16.4	5	2.03
Disagree	1	16.4	4	16.4	7	16.4	12	4.88
Average	12	16.4	10	16.4	21	16.4	43	17.48
Agree	32	16.4	37	16.4	27	16.4	96	39.02
Strongly agree	36	16.4	29	16.4	25	16.4	90	36.59
H_0 is rejected for $p < 0.01$	$\chi^2 = 68.366$	$p < 0.001$	$\chi^2 = 60.073$	$p < 0.001$	$\chi^2 = 30.683$	$p < 0.001$	246	100.00

- Collect DRA usefulness rating numerical data and organize it into Table 24 labelled RT7.
- Plot Table 24 (RT7) data into a bar graph representation Fig. 12 labelled RF7.
- Calculate the statistical values AAF and AAP from Table 24 (RT7) data:
 - AAF: determines the frequency of participants that consider DRA useful.
 - AAP: determines the percentage of participants that consider DRA is useful.

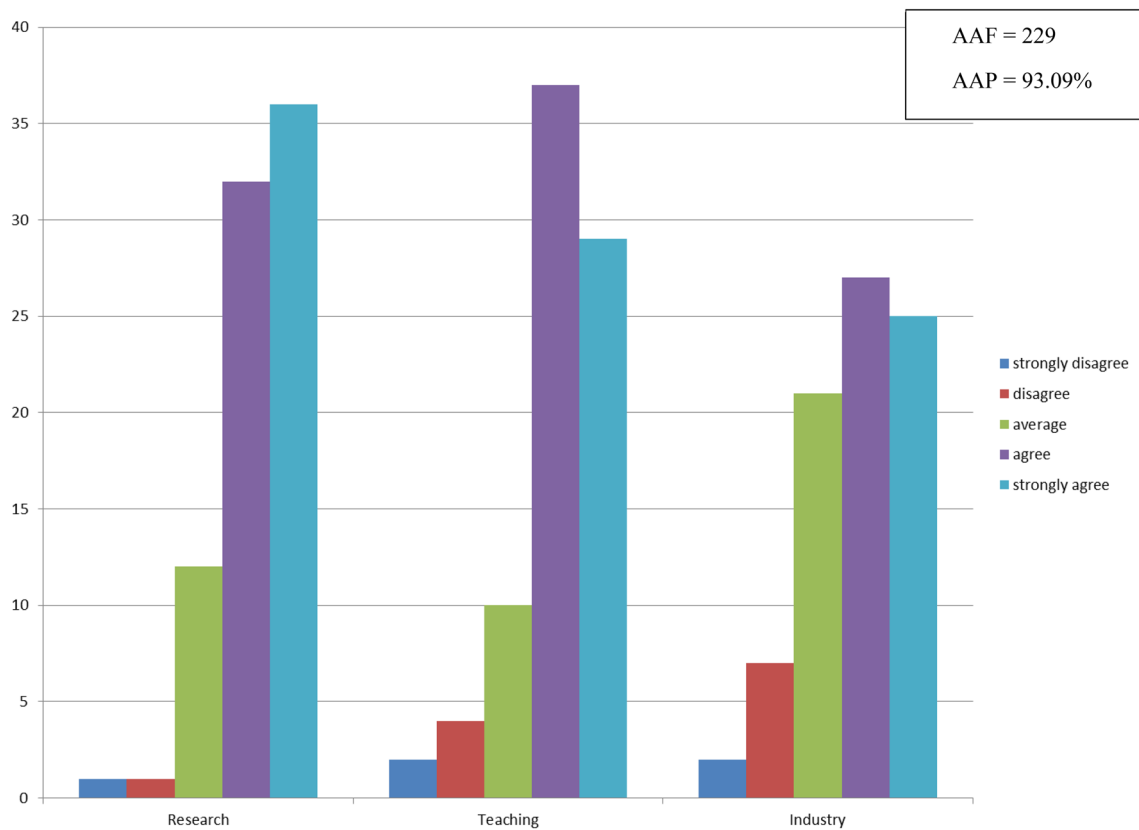


Fig. 12 DRA usefulness ratings graph (RF7)

- Calculate the Goodness of Fit χ^2 and p value for the test variables (teaching, industry, research) at a critical value $\alpha = 0.01$.

If $p < \alpha$ then the null hypothesis H_0 is rejected, H_1 is accepted.

H_0 : There is no association between the test variable (usefulness for teaching, industry, and research) and DRA design models.

H_1 : There is an association between the test variables, and DRA positively meets the test criteria.

- Results summary are labelled as RS [5]. RS [5] presents the analysis of the DRA usefulness feedback data.

RS7: Results Review Table 24 (RT7) statistical analysis results of (Q7-Set) responses can be interpreted as follows:

- AAF = 229 out of total = 246 responses indicate that most of the participants agreed that DRA is useful for teaching, research, and at the industry level.

- AAP = 93.09% indicates that there is a high percentage of participants that agree with the DRA usefulness for teaching, research, and industry.
- The p value for the test variables:
 - Research p value is set at $0.001 < \alpha = 0.01$. This result means that H_0 is rejected, H_1 is accepted, and DRA models are related to usefulness for research.
 - Teaching p value is set at $0.001 < \alpha = 0.01$. This result means that H_0 is rejected, H_1 is accepted, and DRA models are related to the usefulness of teaching.
 - Industry p value is set at $0.001 < \alpha = 0.01$. This result means that H_0 is rejected, H_1 is accepted, and DRA models are related to the usefulness of the industry.

In Table 25, there is a total of $T = 50$ related references to the evaluation criteria (see Table 2). Table 23 criteria frequency distribution is mapped into Table 25 and interpreted as follows:

The participants consider DRA models useful (28.00%), relevant (26.00%), and essential (24.00%). DRA design is generic (10.00%), but it provides sufficient coverage for

Table 25 DRA usefulness categories frequencies

$T=50$	Generalization	Usefulness	Coverage	Relevance	Importance
Frequency	5	14	6	13	12
Percentage	10.00	28.00	12.00	26.00	24.00

industry needs (12.00%). Overall, it can be suggested based on the analysis of results that DRA meets the evaluation criteria set in Table 2. However, participants considered DRA more from usefulness, relevance, and importance perspectives when compared to generalization and coverage. Figure 12 adds further visual evidence that the participants consider DRA useful.

DRA Overall Feedback and Rating This section presents the evaluation of the participants' overall feedbacks and ratings about the DRA. The evaluation process is as follows:

- Collect and map the feedbacks provided about the DRA into Table 26.
- Analyze Table 26 feedback based on the frequency of evaluation criteria (see Table 2) relationship with the DRA aspects in the text using the cross-examination method.
- Collect DRA overall rating and arrange it as numerical data into Table 27 labelled RT8.
- Plot Table 27 (RT8) data into a bar graph representation Fig. 13 labelled RF8.
- Calculate the statistical value AAP from Table 27 (RT8) data.
- AAP: determines the frequency of participants satisfied with the DRA overall.
- Results review labelled RS [1]. RS [1] presents an analysis of the results data.

RS8: Results Summary The numerical data in Table 27 (RT8) produced a principal statistical value based on participants' overall responses. The DRA overall questionnaire Q9-Set showed that:

- The participants consider the DRA satisfactory at $AAP = 77\%$.

- In Table 26, there is a total of $T=22$ related references to the evaluation criteria. Table 25 criteria appearance frequency distribution is mapped into Table 28.

Table 28 shows that participants seem to consider the DRA useful (27.27%), and relevant (45.45%). This result indicates that participants seem to think the DRA relevant to the industry and may be useful for their organizations' contexts. Overall, it can be suggested based on the analysis of results that DRA meets the evaluation criteria set in Table 2. Figure 13 adds further insights that overall, the participants consider DRA useful and relevant.

Key Insights: Summary and Analysis

This section presents the summary and analysis of the DRA empirical evaluation results. The purpose of this investigation is to evaluate the DRA using the industry case study and a field survey. The evaluation data is used to indicate the relationship between the DRA models and the evaluation criteria (Table 2). There are two types of indicative measures designed to determine the significance of the DRA.

- The quantitative indicator matrix (QIM).
- The qualitative evaluator matrix (QEM).

The QIM and QEM aim to provide indicative measurements and qualitative review to verify that the DRA meets the evaluation criteria (Table 2). To complement the quantitative analysis, qualitative feedback was collected and analyzed for rich results and insights, including ideas for further improvement.

The Quantitative Indicator Matrix (QIM)

The quantitative indicator matrix QIM (Table 29) is a collective of the numerical data results reported in the industry survey analysis. The collected data are the χ^2 test, AAF, and AAP (see Eqs. 1, 2, and 3) in statistics report cards Card [3, 6, 8, 10, 13, 59] (see Fig. 5).

The QI formula is an average value that indicates the significance of DRA at a probability above 75%. The QI formula is described in (4) as follows:

QI definition: The Quantitative Indicator (QI) is the average probability (AAP) for every table source in chapter 5, the QIM table (see Table 29).

$$\text{The probability of a source table is : } [\text{Probability } (X) \text{ or } P(X)] = AAP[\text{Table}_{(\text{index})}] \quad (4)$$

Table 26 Q9-set overall feedback analysis

Criteria	Overall feedbacks: participants quotes	Interpretation
Usefulness Coverage Relevance	Overall, it seems very well thought-out and could be extremely useful in practice. However, it is fairly complex and might not be ideal for a newcomer. Perhaps some of the graphics in this form could be further simplified to present DRA's ideas in a cleaner way. I strongly agreed to the "provide enough components" statements; are there too many components? Could it be made simpler? Aside from this, the DRA framework certainly makes sense	DRA is very well thought is extremely useful in practice I strongly agreed to the "provide enough components." DRA framework certainly makes sense
Relevance Usefulness	This work illustrates how the DevOps methodology can be applied to IoT development in a multi-cloud environment. It will be useful for students. It can evolve into a practical solution	DevOps applied to IoT in multi-cloud A useful, practical solution
Usefulness Relevance	I'm afraid it's just not the kind of thing relevant to my work, but I can see how it might be useful to someone starting up a DevOps, multi-cloud, IoT project	Useful for DevOps, multi-cloud IoT
Relevance	I am not technical, so a lot of what is being proposed seemed practical from my view. How would this work with mainframes? I deal with the cultural aside of organisations so while this looks pretty good on paper, I wonder how practical it would be to implement. We get involved a lot in implementing transformations. The people side of things is the hardest. Before we even attempt to introduce technical changed, e.g. dev-ops plus CI/CD, we need to set the organisation up to be able to adapt to the new changes that are about to come. Culture is sometimes the biggest impediment to a successful transformation as you cannot change the culture without first changing the organisation, then this becomes a question of "what is your appetite to change". My point is that this needs to happen prior to introducing any change to an organisation's technical practices and approach if you want it to be successful. Perhaps this is a given as a prerequisite, if not then it should be unless of course, it is a start-up	DRA seems practical
Coverage Usefulness	I really appreciated the breakdown of the theory behind the DevOps culture; it's very good to explain for teaching reasons. In the enterprise world, most of the companies already take it as a practice and its consolidated and evolved further with a "You Build It You Run It" paradigm where operations are part of development, including maintenance and incident management. This is somewhat new to developers to respond to incidents on shifts, without delegating to an ops team designated for maintenance only. This includes the adoption of paging tools (such as OpsGenie or Pager-Duty) to be able to manage an on-call roster for 24/7 response. In the enterprise world, this framework would be separate into two parts, one being technical implementation, and the other one being organization and practices. Architects define the practices and the required organization, while engineers look deeper into the design and implementation of the technical part of the software	DevOps Culture Very good for teaching reasons In the enterprise world, this framework would be separate in two parts: Implementation and Practice
Usefulness Coverage	Great idea about organizing the structure for an IoT project, the main reason for projects failures of IoT is the lack of structure	IT projects Organising the DRA project for IoT
Generalisations	I feel this is a more general architectural reference than specific to IoT devices. I feel there are further unsolved challenges there Self-remediation, infrastructure provisioning should be a part of the framework	General architecture DRA allows retrospective approach for DevOps team
Relevance	Looks effective and definitely have invested a lot of efforts and work but utmost wonderfully presented Many of Industries are already trying to get on these paths. Keep up the good work	Effective Industries are already trying to get on this path
Importance	It will be great automated IoT deploy tool if you keep work on it. It would be good if add a case study of build failure. Sometimes build fails it without reasons	Automation

Table 26 (continued)

Criteria	Overall feedbacks: participants quotes	Interpretation
Usefulness	Happy to see more research like this. The reason I put average for the usefulness of this framework in the industry is that technological innovation happens every day, and new and better tools are constantly introduced into the DevOps space. Therefore some of the tools mentioned in this framework could be replaced and become irrelevant at some point, maybe soon. With that said, I agree the concept remains valid and could be very helpful	Useful DevOps Practices Appropriate Significant concepts
Relevance	This is an excellent project. Brilliant	Excellent project
Relevance	Thanks for sharing your research with me. Unfortunately, I don't have the technical experience to provide more thorough and valuable feedback. But at a conceptual level, it all makes sense. Thanks again, and best of luck!	conceptual level all makes sense
Relevance	Nice project, I never imagined using DevOps for IoT	DevOps for IoT
Relevance	Very interesting, but I think that one should consider GitLab CI/CD as well	Interesting
Relevance Importance	I have learnt a lot within using the DRA approach, and it has enabled me to better understand the way code works and also how these clouds are integrated together	Framework Heterogeneous

Table 27 Survey overall feedback rating (RT8)

On a scale of 1–5. Please provide an overall rating for the DRA framework	
Strongly disagree	1.00
Disagree	2.00
Average	11.00
Agree	46.00
Strongly agree	22.00

$QI = \sum [P(X)]_{\text{row}[i]} / \text{Count}(\text{row}[i])$; where (i = the row index in QIM and X = participants score ≥ 3).

QI objective: Determine the average probability of participants' likely to agree with the data results from source tables (Table_(index)). It indicates that the DRA models meet the evaluation criteria positively.

Specific QI condition: Participants agree with table results: if [AAF > 75%, p value < 0.01].

QI Result: $QI = \sum [P(X) > 75\%]_{\text{row}[i]} / \text{Count}(\text{row}[i])$ [Indicates the average probability of participants scoring ≥ 3 in a particular survey table].

Conclusion: $QI > 75\%$ then DRA meets the evaluation criteria positively.

Data Source: Tables 6, 7, 9, 10, 12, 13, 15, 16, 18, 19, 21, 22 and 24

The QIM (Table 29) aims to determine that there is a probability of 75% or above that the participants agree that the DRA meets the evaluation survey criteria in Table 2. Table 29 generates a probability indicator QI that determines the probability of a participant scoring three or above in the rating table (Table 1) in the survey.

Review $QI = 93.56\%$ indicates that the cohort of participants in the surveys seem to agree that DRA is significant at a condition [Probability > 75% and p value < 0.001]. Hence, it can be concluded that the participants seem to consider that the DRA framework is fit for its intended purposes. Overall, it can be suggested based on the (QIM) analysis of results that DRA meets the evaluation criteria set in Table 2.

The Qualitative Evaluator Matrix (QEM)

The quantitative evaluator matrix (QEM) is a collective of the feedback data gathered in “Industry case study” and “Industry survey”. Three data sources are feeding into the Qualitative Evaluator Matrix QEM (Table 30). The data are acquired from Tables 3, 23 and 26. To complement the quantitative analysis in QIM, the QEM feedback qualitative feedback was analyzed for rich results and insights support that the DRA meets the evaluation criteria (see Table 2). QEM (Table 30) shows the relationships between the evaluation criteria and the expert's feedback.

Review QEM (Table 30) indicates that the participants agree that:

- DRA seems instantiable and easy to implement using various tools and technologies
- DRA seems flexible and reconfigurable with other technology stacks
- DRA includes DevOps tools integration that enables automated deployment to multi-cloud
- DRA is based on high-level modelling that supports DevOps concepts and practices

On a scale of 1 to 5 . Please provide overall rating for the DRA framework

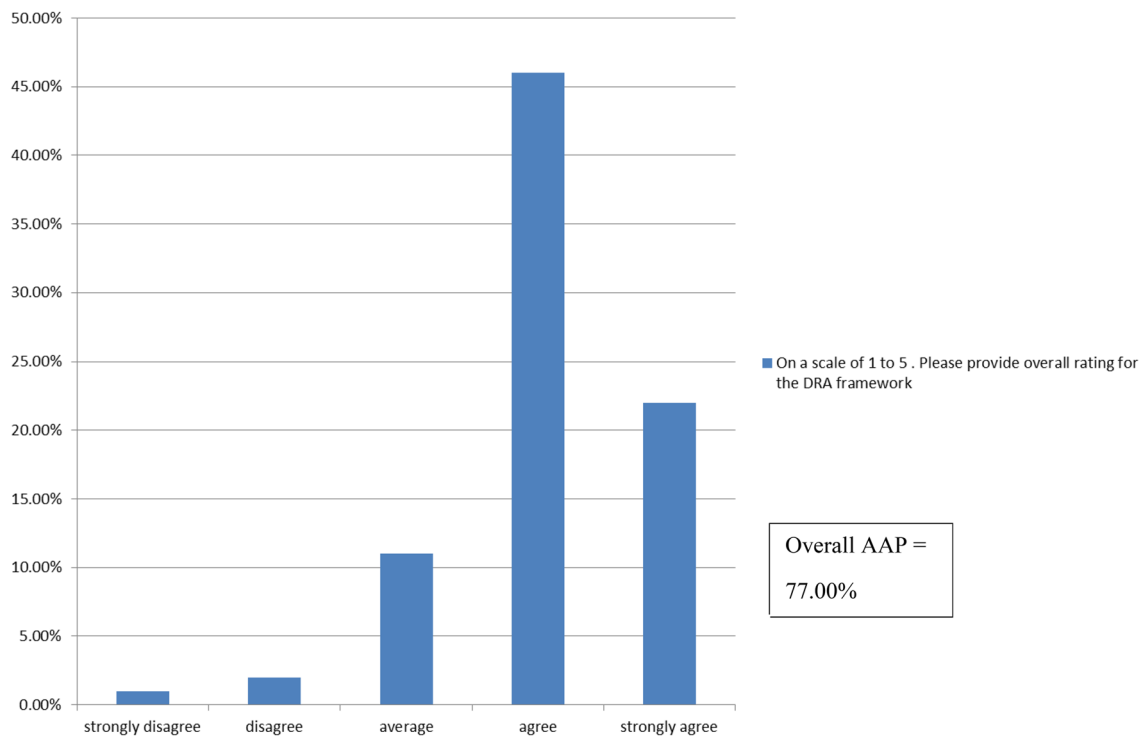


Fig. 13 Survey overall feedback graph (RF8)

Table 28 DRA overall criteria occurrences

T=22	Generalisation	Usefulness	Coverage	Relevance	Importance
Frequency	1	6	3	10	2
Percentage	4.55	27.27	13.64	45.45	9.09

- DRA seems to enable DevOps concepts, multi-cloud services and support IoT process.
- DRA seems to support DevOps culture and human factor
- DRA is a comprehensive architecture that may support digital transformation
- DRA seems to provide new knowledgebase about DevOps approach adoption.
- DRA seems to provide a fast and straightforward path to software production in Agile.
- DRA is generic and applicable to a class of situations.

Future Scope

This section evaluates the participants’ responses regarding suggested improvements to the DRA at the industry level. The feedback was in response to the Q8-Set.

Feedback to the Q8-Set provided vital suggestions to improve further the DRA, as well as valuable ideas that may be considered as future research projects or DRA upgrades. The evaluation data results for the Q8-Set are presented in Table 31, which is organized as follows: the participants’ suggestions column, which contains the participants’ feedback and comments; and the interpretations column, which includes the researcher’s answers to suggestions. In this column, the researcher identifies the key ideas considered for the possible future scope of the research.

Discussion

Most recently, DevOps has emerged to deal with this vital concern by enabling the integration of development and operations abilities to complement the current agile approaches [1, 2]. DevOps approach provides developers with concepts, practices, and tools to enable automation, continuous integration, and fast deployment and delivery on the cloud [5, 58]. This paper discusses the evaluation of the proposed DRA framework to deploy IoT-applications to multi-cloud using DevOps [8]. The DRA was constructed in previous research [8] using a well-known design science

Table 29 Quantitative Indicator Matrix (QIM)

Table _(index)	AAP	AAF	Coverage <i>p</i> value	Relevance <i>p</i> value	Importance <i>p</i> value	Useful for research	Useful for teaching	Useful for industry	<i>P</i> (<i>X</i>)
Table 6 Table 7	370	90.25%	<0.001	<0.001	<0.001				90.25%
Table 9 Table 10	382	93.17%	<0.001	<0.001	<0.001				93.17%
Table 12 Table 13	386	94.14%	<0.001	<0.001	<0.001				94.14%
Table 15 Table 16	711	96.33%	<0.001	<0.001	<0.001				96.33%
Table 18 Table 19	380	92.68%	<0.001	<0.001	<0.001				92.68%
Table 21 Table 22	625	95.28%	<0.001	<0.001	<0.001				95.28%
Table 5.39	229	93.09%				<0.001	<0.001	<0.001	93.09%
$QI = \sum [P(X) > 75\%]_{\text{row}[i]} / \text{Count}(\text{row}[i]) = 93.56\%$									

research methodology [12]. The DRA architectural model provides agile teams with real-time monitoring, team collaboration, automated deployment to multi-cloud capability to the application at run-time using the DevOps approach and cloud-tools and services.

The DRA seems to offer developers a research-based and practical architecture driven in-depth approach to plan, analyze, architect, and implement the DevOps automation environment to support the deployment stage of software engineering for multi-cloud IoT-applications. To provide sufficient proof of the applicability and novelty of the DRA, this paper presents the results of the empirical evaluation of the DRA architectural models [13, 59], which is an essential step for developing a novel Agile-DevOps theory. The empirical evaluation is composed of two main sections: (1) industry case study, (2) online field survey.

First, the industry case study used a CST (Link) to provide instructive guidelines for DRA implementation at the CPF organization. The feedback obtained from the DevOps engineer expert at CPF were analyzed in Table 3 and were reported in Table 4. Table 4 results indicate that DRA is useful, reusable, and covers the organization's needs.

Second, the industry survey (Link) used in this paper was offered online using the author's LinkedIn to industry experts from organizations located locally and internationally. The survey data analysis is composed of two phases: quantitative data analysis and qualitative data analysis. The RT [index] and CT[index] tables produced key statistical values (AAP, AAF, and $\chi^2 p$ value).

Third, an overall summary and analysis of the results were presented for crucial insights. The QIM (Table 27) produced a probability $QI = 93.56\%$, which means that participants in the case study and survey agree that DRA is fit for its intended purposes. The QEM (see Table 30) demonstrates

that the participants consider the DRA architectural models fit for its intended purposes. The QEM indicate that the DRA positively meets the evaluation criteria in Table 2. Hence, based on this evaluation results, it can be suggested that the DRA framework is a practical and applicable approach for IoT-application deployment to multi-cloud. Also, it has been founded during this research that the new DRA, the CI-Broker concept, may also be used for other non-IoT applications. The empirical evaluation results provide sufficient information about the usefulness and applicability of the DevOps adoption for software development.

This study does not claim to offer complete knowledge of the integration of the three contexts [DevOps, Multi-Cloud, and IoT]. However, it is anticipated that the results of this study will provide sufficient consolidated and synthesized information and insights to practitioners and researchers and enable them to make informed decisions about the adoption of DevOps for IoT on the cloud. Further, it will provide a strong foundation, grounded on empirical evaluation results, for developing the theories in this vital area of research.

Conclusion

The integration of the Development and Operations is itself a complex subject, and it becomes more challenging when it is associated with the emerging IoT and multi-cloud contexts. This paper presented the evaluation of one such framework, DRA. The proposed DRA has been developed using a well-known design science research methodology.

Table 30 Qualitative evaluator matrix (QEM)

Criteria	Feedback: participants quotes
Importance Reusable Generalisations	“One particular aspect that I think is very important in the DRA framework is the flexibility to choose the instantiations of each component (or indeed have several instantiations). Given the heterogeneous nature of the cloud and IoT environments, I think this is a critical feature”
Usefulness Coverage Relevance	“The models provided are all very useful as they deconstruct and disambiguate what is required in order to deploy any code as a single or multi-cloud application—this is achieved through presenting both abstract and concrete examples and clearly defining steps involved at each stage of the process”
Usefulness Importance	“The continuous integration and automated deployment to multi-cloud is very useful”
Generalisations Coverage	“All of it, it provides a comprehensive overview of what’s needed for an IoT application deployment and management”
Novelty Importance Relevance	“Decentralized logging, cloud hosted CI, deployment to multiple hosting vendors, etc. are all acceptable modern solutions for the demonstrated problem”
Usefulness	“DRA logical model specifications and DRA pipeline instance are very useful in the enterprise world, as they focus on the low-level implementation scheme”
Relevance	“The fact that the entire deployment process is automated and seamless is really nice”
Relevance	Interesting approach to combine DevOps and IoT
Coverage Usefulness Generalisations	“I do like this model for deployment of applications. I think it is quite extensive and applicable to developers, (ops) and IT management including project management”
Usefulness	“It provides a framework that I’m pretty sure would be invaluable for people that don’t know all the components and would like to implement it”
Coverage	“Correctly identifies the benefits of DevOps”
Importance Reusable	“Very appropriate tools and real use case implementation”
Relevance	“It gives a clear idea of a path to production that could be used in IoT development process”
Usefulness	“Useful for IoT cloud solutions”
Importance Usefulness	“The DevOps section is very good as all the tools are perfectly used. As it will make good automation in architecture”
Generalisations Relevance Usefulness	“Having a high-level view of specifications, logic, modelling, and behaviour of the whole system is paramount to a solid implementation of the pipeline between IoT and Multi-Cloud”
Relevance Importance	“It’s good that you’re creating a platform on the multi-cloud level. The idea of Automating end to end pipeline is a good thing.”
Usefulness	“I am really impressed with DRA I think the IoT environment are missing some DevOps structure like that provided by DRA, so it is very useful to guide teams to have a simple workflow to implement DevOps in IoT projects”
Relevance Usefulness Reusable	“The value proposition that is to enable IoT architecture by the DevOps efforts in order to start using this amazing technology for an application like smart cities or even the set a cloud architecture for a smart house system”
Importance Usefulness	“Definitely automation. When IoT makes to the top, it will be so hard to test things manually, and those things are going to be interacting directly and physically with users... So the automation process it’s going to be a very good friend for the UX of anything related IoT”
Usefulness Relevance Importance	“This can be a useful reference document for IoT based Apps deployment with DevOps culture in the team process and tool-set”
Generalisations	“I thought it was a good overview of how DevOps can be applied with IoT”
Importance	“It is a good high-level design for the IoT cloud app deployment workflow”
Relevance Importance	“Agility, CI/CD, automation, Speed”
Importance Relevance	“Allowing users to interact with IoT devices remotely using cloud services. The ability to track usage through the use of Paper-trail and automation of IoT device by using the cloud”
Relevance Importance Coverage Novelty	“Perfect pick up as industry is facing these issues as to how to integrate or implement DevOps transformation when it comes to Cloud and IoT related Apps. In a nutshell its part of Digital Transformation which is one of future in Industry”

Table 30 (continued)

Criteria	Feedback: participants quotes
Usefulness Coverage Relevance	“Overall, it seems very well thought-out and could be extremely useful in practice. However, it is fairly complex and might not be ideal for a newcomer. Perhaps some of the graphics in this form could be further simplified to present DRA’s ideas in a cleaner way. I strongly agreed to the “provide enough components” statements; are there too many components? Could it be made simpler? Aside from this, the DRA framework certainly makes sense”
Coverage Usefulness	“I really appreciated the breakdown of the theory behind the DevOps culture; it’s very good to explain for teaching reasons. In the enterprise world, most of the companies already take it as a practice and its consolidated and evolved further with a “You Build It You Run It” paradigm where operations are part of development, including maintenance and incident management ...etc.”
Relevance	“Looks effective and definitely have invested a lot of efforts and work but utmost wonderfully presented Many of Industries are already trying to get on these paths. Keep up the good work”
Usefulness	“Happy to see more research like this. The reason I put average for the usefulness of this framework in the industry is that technological innovation happens every day, and new and better tools are constantly introduced into the DevOps space. Therefore some of the tools mentioned in this framework could be replaced and become irrelevant at some point, maybe soon. With that said, I agree the concept remains valid and could be very helpful”
Usefulness	“Tools used in Operation model pipeline are industry used tools and are an excellent choice for the DRA Operation Mode Pipeline”
Reusable	“Configuration template is easy to use and can be replicated”
Coverage Usefulness	“DRA framework would help organisations understanding DevOps methodologies and agile application deployment and delivery”
Generalisations	“DRA is applicable and is fit for purpose to set up the DevOps multi-cloud”
Usefulness	“DRA is general in the sense that it is not fixed to one situation or environment and can adapt to different situations and be used with different technology stacks as appropriate to the situation. Thus DRA is applicable to a class of problem situations and is applicable to several instantiations”
Novelty	“DRA offers new knowledge, which has not been discussed before in the form of complex DevOps for Multi-cloud and IoT”
Coverage	“DRA models seem to provide sufficient explanation about the elements and their relationships as a “design knowledge”, which can be used or re-used for a class of a problem addressed in this work”
Reusable	“My overall feedback is that DRA can be successfully instantiated for the similar research lab environment needs for the deployment of IoT applications using multi-cloud. Overall DRA is fit for purpose”

Table 31 Future scope and suggested improvements

Future scope
An important aspect that should be considered is governance. Although it does not play a role in automation, it is a very important component in real businesses. Could be an improvement for future versions
Consider having a step to create and monitoring error logs from production devices
Security and Metrics are an important piece of DevOps
Security for infrastructure as well as application and deployment speed improvements
DevSecOps or DevQAOps can be included as part of detailing. E.g., TDD/BDD Framework or Security-related tools can be added

To conclude the research, this paper focused on the evaluation phase of the DRA. Thus, the DRA has been evaluated using the industry case study and survey. Based on the evaluation results, it was determined that DRA is an appropriate framework for architecting and implementing the DevOps for the automated deployment of IoT applications to the multi-cloud. This paper is a crucial contribution to the development of a theory in this vital area of research. Further, this research demonstrates how to systematically evaluate a complex artifact, such as the DRA, using the DSR evaluation criteria. Finally, this research can be extended to include a future investigation into the areas of DevSecOps security and DataOps.

Appendix

The evaluation of the DRA is composed of industry case study and industry field survey. The tools and templates used in the evaluation and data collection are stored on CloudStor. Please refer to the following web-links for more information:

- Industry case study template (CST): <http://tiny.cc/nyu9iz>
- Industry Case Study data: <http://tiny.cc/85u9iz>
- Industry Field Survey: <http://tiny.cc/99dldz>

- Industry Survey Data: <http://tiny.cc/w8u9iz>
- Survey Participants' Distribution: <http://tiny.cc/u9u9iz>
- DRA Application Demo Video (Used in CST and Survey): <https://youtu.be/JN38xS27ek0>
- DRA Presentation Slides (Used in CST and Survey): <http://tiny.cc/pcv9iz>

Please note: Personal information about the participants in the industry case study and the industry field survey were kept anonymous.

Funding The authors wish to thank the Australian Government Research Training Program (RTP) for providing the funding for this research project. We also extend experts from academic and industry who provided cortical and constructive feedback to improve the quality and outcomes of this research.

Compliance with Ethical Standards

Conflict of interest Conceptualization, virtualization, methodology, formal analysis, investigation, and validation were prepared and conducted by Dr Georges Bou Ghantous. Dr Georges Bou Ghantous developed the software for the proof of concept. Dr Georges Bou Ghantous acquired resources and funding. The original draft of the manuscript was prepared and written by Dr Georges Bou Ghantous. All authors managed the project. A. Prof Asif Qumer Gill completed the manuscript writing (review and editing). A. Prof Asif Qumer Gill supervised the research project.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Alzoubi YI, Gill AQ, Al-Ani A. Distributed agile development communication: an agile architecture driven framework. *JSW*. 2015;10(6):681–94.
2. Bai X, Pei D, Li M, Li S. The DevOps lab platform for managing diversified projects in educating agile software engineering. In: 2018 IEEE frontiers in education conference (FIE), San Jose, CA, USA; 2018, pp. 1–5. <https://doi.org/10.1109/FIE.2018.8658817>.
3. Bou Ghantous G, Gill A. DevOps: Concepts, practices, tools, benefits and challenges. PACIS2017. Pacific Asia Conference on Information Systems 2017, Sep 11. AISel <https://aisel.aisnet.org/pacis2017/>.
4. Chang CK. Agile, Continuous Integration, and DevOps. In: 2019 IEEE 43rd annual computer software and applications conference (COMPSAC), vol. 1. IEEE; 2019. pp. 211.
5. Rajkumar M, Pole AK, Adige VS, Mahanta P. DevOps culture and its impact on cloud delivery and software development. In: 2016 international conference on advances in computing, communication, & automation (ICACCA) (Spring), Dehradun; 2016. pp. 1–6. <https://doi.org/10.1109/ICACCA.2016.7578902>
6. Moore J, Kortuem G, Smith A, Chowdhury N, Cavero J, Gooch D. DevOps for the Urban IoT. *Urb-IoT'16*, May 24–25, 2016, Tokyo, Japan. 2016. <https://doi.org/10.1145/2962735.2962747>
7. Anwar MJ, Gill AQ. A review of the seven modelling approaches for digital ecosystem architecture. In: 2019 IEEE 21st conference on business informatics (CBI), vol. 1. IEEE; 2019. pp. 94–103.
8. Bou Ghantous G, Gill AQ. DevOps reference architecture for multi-cloud IoT applications. In: 2018 IEEE 20th conference on business informatics (CBI), Vienna, 2018, pp. 158–167. <https://doi.org/10.1109/CBI.2018.00026>.
9. Wetzinger J, Andrikopoulos V, Leymann F, Strauch S. Middleware-oriented deployment automation for cloud applications. *IEEE Trans Cloud Comput*. 2018;6(4):1054–66. <https://doi.org/10.1109/TCC.2016.2535325>.
10. Tao M, Zuo J, Liu Z, Castiglione A, Palmieri F. Multi-layer cloud architectural model and ontology-based security service framework for IoT-based smart homes. *Future Gener Comput Syst*. 2018;78:1040–51.
11. Dreibholz T, Mazumdar S, Zahid F, Taherkordi A, Gran EG. Mobile edge as part of the multi-cloud ecosystem: a performance study. In: 2019 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), Pavia, Italy; 2019. pp. 59–66. <https://doi.org/10.1109/EMDPD.2019.8671599>.
12. Peffers K, Tuunanen T, Rothenberger M, Chatterjee S. A design science research methodology for information systems research. *J Manag Inf Syst*. 2007;24(3):45–77.
13. Prat N, Comyn-Wattiau I, Akoka J. Artifact evaluation in information systems design-science research—a holistic view. In: PACIS; 2014. p. 23.
14. Carvalho JÁ. Validation criteria for the outcomes of design research. In: Pre-ECIS workshop on IT artefact design and work practice intervention; 2012.
15. McCarthy MA, Herger LM, Khan SM, Belgodere BM (2015) Composable DevOps: automated ontology-based DevOps maturity analysis. In: In 2015 IEEE international conference on services computing, IEEE, pp. 600–607.
16. Artač M, Borovšak T, Di Nitto E, Guerriero M, Tamburri DA. Model-driven continuous deployment for quality DevOps. In: Proceedings of the 2nd international workshop on quality-aware DevOps. New York: ACM; 2016. pp. 40–41.
17. Samarawickrama, Saliya Sajith, and Indika Perera. Continuous Scrum: A Framework to Enhance Scrum with DevOps. 2017 Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer), 2017. <https://doi.org/10.1109/ictcr.2017.8257808>.
18. Karapantelakis A, Liang H, Wang K, Vandikas K, Inam R, Fersman E, Mulas-Viela I, Seyvet N, Giannokostas V. DevOps for IoT applications using cellular networks and cloud. In: 2016 IEEE 4th international conference on future internet of things and cloud (FiCloud). IEEE; 2016. pp. 340–347.
19. Wetzinger J, Breitenbücher U, Kopp O, Leymann F. Streamlining DevOps automation for cloud applications using TOSCA as standardized metamodel. *Future Gener Comput Syst*. 2016;56:317–32.
20. Soni M. End to end automation on cloud with build pipeline: the case for DevOps in insurance industry, continuous integration, continuous testing, and continuous delivery. In: IEEE international conference on cloud computing in emerging markets 2015. Piscataway: IEEE; 2015. pp. 85–89.
21. Syed MH, Fernandez EB. Cloud ecosystems support for internet of things and DevOps using patterns. In: 2016 IEEE first

- international conference on internet-of-things design and implementation (IoTDI). Berlin; 2016. pp. 301–304. <https://doi.org/10.1109/IoTDL.2015.31>.
22. Douzis K, Sotiriadis S, Petrakis EG, Amza C. Modular and generic IoT management on the cloud. *Future Gener Comput Syst.* 2018;78:369–78.
 23. Ferry N, Chauvel F, Song H, Rossini A, Lushpenko M, Solberg A. CloudMF: model-driven management of multi-cloud applications. *ACM Trans Internet Technol.* 2018;18(2):16.
 24. Yonezawa T, Ito T, Nakazawa J, Tokuda H. Soxfire: a universal sensor network system for sharing social big sensor data in smart cities. In: *Proceedings of the 2nd international workshop on smart.* New York: ACM; 2016. p. 2.
 25. Cavalcante E, Pereira J, Alves MP, Maia P, Moura R, Batista T, Delicato FC, Pires PF. On the interplay of Internet of Things and cloud computing: a systematic mapping study. *Comput Commun.* 2016;89:17–33.
 26. Khakimov A, Muthanna A, Kirichek R, Koucheryavy A, Muthanna MSA. Investigation of methods for remote control IoT-devices based on cloud platforms and different interaction protocols. In: *IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering 2017.* Piscataway: IEEE; 2017. pp. 160–163.
 27. Yaqoob I, Ahmed E, Hashem IAT, Ahmed AIA, Gani A, Imran M, Guizani M. Internet of Things architecture: recent advances, taxonomy, requirements, and open challenges. *IEEE Wirel Commun.* 2017;24(3):10–6.
 28. Gutiérrez-Madroñal L, Medina-Bulo I, Domínguez-Jiménez JJ. IoT-TEG: test event generator system. *J Syst Softw.* 2018;137:784–803.
 29. Botta A, De Donato W, Persico V, Pescapé A. Integration of cloud computing and internet of things: a survey. *Future Gener Comput Syst.* 2016;56:684–700.
 30. Leite J, Batista T, Oquendo F. Architecting IoT applications with SysADL. In: *IEEE international conference on software architecture workshops 2017.* Piscataway: IEEE; 2017. pp. 92–99.
 31. Kolios P, Panayiotou C, Ellinas G, Polycarpou M. Data-driven event triggering for IoT applications. *IEEE Internet Things J.* 2016;3(6):1146–58.
 32. Yasaki K, Ito H, Nimura K. Dynamic reconfigurable wireless connection between smartphone and gateway. In: *IEEE 39th annual computer software and applications conference 2015, vol. 3.* Piscataway: IEEE; 2015. pp. 228–233.
 33. Babovic ZB, Protic J, Milutinovic V. Web performance evaluation for the internet of things applications. *IEEE Access.* 2016;4:6974–92.
 34. Alkhalil A, Ramadan RA. IoT data provenance implementation challenges. *Procedia Comput Sci.* 2017;109:1134–9.
 35. Sen S. Context-aware energy-efficient communication for IoT sensor nodes. In: *53rd ACM/EDAC/IEEE design automation conference 2016.* Piscataway, IEEE; 2016. pp. 1–6.
 36. Jula A, Sundararajan E, Othman Z. Cloud computing service composition: a systematic literature review. *Expert Syst Appl.* 2014;41(8):3809–24.
 37. Sun D, Fu M, Zhu L, Li G, Lu Q. Non-intrusive anomaly detection with streaming performance metrics and logs for DevOps in public clouds: a case study in AWS. *IEEE Trans Emerg Top Comput.* 2016;4(2):278–89. <https://doi.org/10.1109/TETC.2016.2520883>.
 38. Nayyar A. *Handbook of cloud computing: basic to advance research on the concepts and design of cloud computing.* Delhi: BPB Publications; 2019.
 39. Singh P, Gupta P, Jyoti K, Nayyar A. Research on auto-scaling of web applications in cloud: survey, trends and future directions. *Scalable Comput Pract Exp.* 2019;20(2):399–432.
 40. Li W, Santos I, Delicato FC, Pires PF, Pirmez L, Wei W, Song H, Zomaya A, Khan S. System modeling and performance evaluation of a three-tier cloud of things. *Future Gener Comput Syst.* 2017;70:104–25.
 41. Domaschka J, Griesinger F, Baur D, Rossini A. Beyond mere application structure thoughts on the future of cloud orchestration tools. *Procedia Comput Sci.* 2015;68:151–62.
 42. Willnecker F, Krcmar H. Multi-objective optimization of deployment topologies for distributed applications. *ACM Trans Internet Technol.* 2018;18(2):21.
 43. Kritikos K, Plexousakis D. Multi-cloud application design through cloud service composition. In: *IEEE 8th International Conference on Cloud Computing 2015.* Piscataway: IEEE; 2015. pp. 686–693.
 44. Yasrab R, Gu N. Multi-cloud PaaS Architecture (MCPA): a solution to cloud lock-in. In: *3rd international conference on information science and control engineering 2016.* Piscataway: IEEE; 2016. pp. 473–477.
 45. Yang C, Shen W, Lin T, Wang X. A hybrid framework for integrating multiple manufacturing clouds. *Int J Adv Manuf Technol.* 2016;86(1–4):895–911.
 46. Slawik M, Blanchet C, Demchenko Y, Turkmen F, Ilyushkin A, de Laat C, Loomis C. CYCLONE: the multi-cloud middleware stack for application deployment and management. In: *IEEE international conference on cloud computing technology and science 2017.* Piscataway: IEEE; 2017. pp. 347–352.
 47. Jamshidi P, Pahl C, Chinenyeze S, Liu X. Cloud migration patterns: a multi-cloud service architecture perspective. *Service-Oriented Computing-ICSOC 2014 Workshops.* Cham: Springer; 2015. p. 6–19.
 48. Shekhar S, Gokhale A. Enabling IoT applications via dynamic cloud-edge resource management. In: *IEEE/ACM second international conference on internet-of-things design and implementation 2017.* Piscataway: IEEE; 2017. pp. 331–332.
 49. Di Martino B, Esposito A. Semantic techniques for multi-cloud applications portability and interoperability. *Procedia Comput Sci.* 2016;97:104–13.
 50. Diekmann C, Naab J, Korsten A, Carle G. Agile network access control in the container age. *IEEE Trans Netw Serv Manag.* 2019;16(1):41–55. <https://doi.org/10.1109/TNSM.2018.2889009>.
 51. Hong Z, Chen W, Huang H, Guo S, Zheng Z. Multi-Hop cooperative computation offloading for industrial IoT-Edge-Cloud computing environments. *IEEE Trans Parallel Distrib Syst.* 2019;30(12):2759–74.
 52. Berger S, Häckel B, Häfner L. Organizing Self-Organizing Systems: A Terminology, Taxonomy, and Reference Model for Entities in Cyber-Physical Production Systems. *Inf Syst Front.* 2019. <https://doi.org/10.1007/s10796-019-09952-8>.
 53. Gregor S, Hevner AR. Positioning and presenting design science research for maximum impact. *MIS Q.* 2013;37(2):337–55.
 54. Bou Ghantous G, Gill AQ. An agile-DevOps reference architecture for teaching enterprise agile. *Int J Learn Teach Educ Res.* 2019;18(7):128–44.
 55. Runeson P, Höst M. Guidelines for conducting and reporting case study research in software engineering. *Empir Softw Eng.* 2009;14(2):131.
 56. Jedlitschka A, Pfahl D. Reporting guidelines for controlled experiments in software engineering. In: *2005 international symposium on empirical software engineering.* IEEE; 2005. pp. 10.
 57. Hyndman RJ. *Quantitative Business Research Methods.* Department of Econometrics and Business Statistics Monash University (Clayton campus). Vic 3800. DBA6000; 2008.
 58. Kao CH. Continuous evaluation for application development on cloud computing environments. In: *2017 international conference*

- on applied system innovation (ICASI), Sapporo, 2017, pp. 1457–1460. <https://doi.org/10.1109/ICASI.2017.7988191>.
59. Venable J, Pries-Heje J, Baskerville R. A comprehensive framework for evaluation in design science research. In: International conference on design science research in information systems. Berlin: Springer; 2012. pp. 423–438.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.