INTERNATIONAL JOURNAL OF ADVANCED STUDIES IN COMPUTER SCIENCE & ENGINEERING
IJASCSE VOLUME 12 ISSUE 4, 2023
04/30/2023

Cloud-native architecture Portability Framework Validation and Implementation using Expert System

Daniel Olabanji1, Tineke Fitch1, Olumuyiwa Matthew1

School of Computing, University of Portsmouth, UK

Abstract: Using Artificial intelligence to solve semi- or ill-structured problems using an algorithm that deploys techno-scientific human experts' approach (Expert System) is a widely used solution. Expert system (ES) provides a programmable methodology solution through instructions provided by intelligence based on human experts. The expert system was used in this study to validate the decision process of the cloud-native architecture portability framework. The cloud-native architecture portability framework is developed to support decision-makers in organizations in making the right decision on porting or migrating either legacy or cloud-based data or applications to cloud-native architecture. The framework, designed and developed from research and expert contributions, was implemented in an expert system to examine its validity. The framework was evaluated through data collected from questionnaires, and the findings show that most respondents agreed with the importance of the framework. Then the evaluated framework was then developed into an expert system to provide a clear path for the stakeholders and the task and user-centred view of the framework. The usability of the designed expert system through the use of the ES-BUILDER shell also shows the usefulness of artificial intelligence in decision-making and information presentation simplification through technology.

Keywords – Expert System, Cloud native architecture, Cloud computing, Validation, Artificial Intelligence

1. INTRODUCTION

Expert systems (ES) is a branch of Artificial Intelligence (AI) that widely uses techno-scientific human expertise to solve semi, or ill-structured problems, where using an algorithm to solve the problem is not guaranteed (Rajabi et al., 2019). An expert system extensively uses a human expert with specialised knowledge to solve the problem. Expert systems represent a programming methodology by which a computer can be instructed to perform tasks which have previously been considered to require the intelligence of a human expert (Matthew et al., 2016). The expert systems have been characterised as an intelligent program that applies knowledge and inference steps to handle complex problems which require essential human knowledge for their solutions field (Feigenbaum & Bond, 1981; Rajabi et al., 2019). Several names have been given to expert systems in research, such as fussy expert systems (Rajabi et al., 2019), knowledge-based systems, and knowledge-based decision-supporting systems (Chung et al., 2016), (Zhang et al., 2022). However, any system that essentially emulates the acquired knowledge and thought process of an expert in any field in making or arriving at a decision and using that decision in solving the problem(s) could be referred to as an expert system. An expert systems are designed to address complex problems by imitating human experts and clarifying the reasoning process of the experts in which knowledge is represented symbolically rather than numerically (Matthew et al., 2016). The expert system can be an essential tool for the organization to supplement decision-makers, improve worker productivity, time and cost saving and preserve and document knowledge (Liebowitz, 1995). The decision about whether to port into cloud-native architecture is much of reasoning rather than the number, making the knowledge-based decision-supporting system a vital system to validate the developed framework.

The cloud-native architecture portability framework developed and design by the author is based on the knowledge acquired from experts in the field of cloud computing and cloud-native architecture. Migration into an advance cloud computing technology can be better achieved through the adoption of cloud-native architecture, and how to achieve it has been shown through research as a complex thing which leads to the design and development of the framework. This paper presents the implementation of the cloud-native architecture portability framework into an ES—the validation processes of both the framework and the ES of the framework.

The rest of the paper is organised as follows. Section 2 provides a brief introduction to cloud-native architecture and portability. Section 3 details the decision process and the framework for the migration into cloud-native architecture. Section 4 will provide a details validation of the framework and the experts' system Section 5 illustrates the integrations of cloud-native architecture portability framework into the expert system. Finally, Section 6 offers some concluding remarks and outlines our future work.

2. CLOUD-NATIVE ARCHITECTURE AND PORTABILITY

Cloud-native is not a new stand-alone term in the cloud computing community. Thus, it is part of the cloud computing paradigm. The term has been used in the early period of the development of cloud computing technology to mean applications developed solely for cloud computing (Andrikopoulos et al., 2013). The progression and improvement in cloud computing through the development of several parts of the technology, the user needs and innovation to achieve all that cloud computing promise to deliver the fundamental characteristic of cloud computing. Cloud-native has evolved from the early idea to mean something different, which will be further discussed in this section. Cloud-Native Computing Foundation (CNCF), the sole convener of the technology, defines cloud-native architecture as a set of technologies that empower organizations to build and run scalable applications in modern, dynamic environments such as public, private, and hybrid clouds. The approach is exemplified by "containers, service meshes, microservices, immutable infrastructure, and declarative APIs"(CNCF, 2019). Furthermore, CNCF asserts that cloud-native architecture enables loosely coupled systems that are resilient, manageable, and observable with robust automation that allows engineers to make high-impact changes frequently and predictably with minimal work. Cloud-native technologies are used to develop applications with services packaged in containers, deployed as microservices, and managed on elastic infrastructures through agile DevOps processes and continuous delivery workflows (Toffetti et al., 2017).

Figure 1: Cloud Native Application Pyramid

Cloud-native architecture is a complete design methodology that complies to deploy and develop applications built with services packaged in containers, deployed as microservices, and managed on elastic infrastructure through agile DevOps processes and continuous delivery workflows as seen in Figure 1. In agreement, (Kratzke & Quint, 2017) discussed that cloud-native deployment or development of applications should agree with the cloud-native's principle, properties, methods and architecture. Cloud-native application (CNA) is then defined as "a distributed, elastic and horizontally scalable system composed of (micro)services which isolate state in a minimum of stateful components (Bajaj et al., 2020).The application and each self-contained deployment unit of that application is designed according to cloud-focused design patterns and operated on a flexible self-service platform" (Kratzke & Quint, 2017). Adopting and migrating to the architecture is complex, and a decision-supporting system is needed.

It is noteworthy that portability in cloud computing is mainly divided into two: Data portability is defined as the ability of cloud consumers to move data objects into or out of a cloud or to use a disk for bulk data transfer. On the other hand, system portability is the ability to move a machine image from one cloud infrastructure or provider to another. It can also mean migrating applications, services and contents from one service provider to another(Liu et al., 2011). An organization that is considering porting into cloud-native architecture face a need to decide on a different area of the migration. Using the framework will eradicate the issues of cost, lack of technical expertise, and complexity of architecture that the framework phases and steps can solve.

3. REVIEW OF THE CLOUD-NATIVE ARCHITECTURE DECISION SUPPORTING FRAMEWORK

The cloud-native architecture migration framework is designed for an organization already using cloud-native architecture or considering the adoption of cloud-native architecture in their businesses. Organizations can use the framework to check their business consideration, needs, benefits and the thoughts put in place before and after porting. The need for a framework based on the decision-making process for an organization is essential. It is a challenging task due to the wide and ever-expanding variety of IaaS service offerings, with the need for more knowledge about the selection criteria in a diverse market that is likely to confuse the customer (Samreen et al., 2020). Decision-making processes include a series of steps, structure the decision problem, access the possible impact of alternatives, determine the decision-makers preferences, and evaluate and compare alternative. The decision process can be used to develop a framework whose core function is to act as an assessment tool for crucial stakeholders when selecting cloud services and to guide decision-makers (Opara-Martins et al., 2017). A regulatory framework for decision-making is one of the guidelines to support effective decision-making processes for the enterprise (Cunningham & Cunningham, 2021) The novel framework is divided into mini steps and tasks for easy understanding and to give a clear path that can provide organizations with the support needed to implement and use the framework. The steps are grouped into four (4) key phases, which are:

- Phase 1: Intention and planning
- Phase 2: Model Selection and Evaluation
- Phase 3: Contract and Execution Implementation
- Phase 4 Testing and Management

The framework was evaluated using data collected from a questionnaire set for experts in cloud computing in which 20 participants participated in the survey. The data collected from the participants show that each phase, steps, and tasks are essential and valuable. The framework was incorporated into an expert system to validate the human expert design and view of the framework. Furthermore, the usage, testing and understanding of the decision support system are further understood through the Expert system.

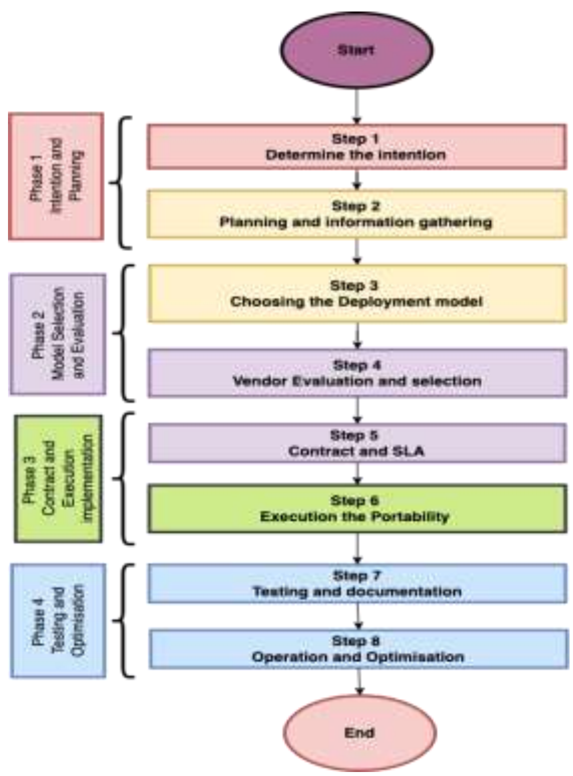

Figure 2: Step-by-step representation of the cloud-native architecture decision supporting framework

4. Evaluation of the framework and Expert system

This section presents the empirical evaluation of the usability and effectiveness of the novel 8-step decision framework which is done through an expert evaluation method based on framework's usability with the help of experts specialized in the subject matter. The expert evaluation method is a method that is used to evaluate parameters of complex objects effectively and make decisions based on them. Experts that understand how to receive information, digest it and provide enhanced information based on experience and knowledge are the right and frequently set of people to be contacted for such evaluation (Poleshchuk, 2018). The expert evaluation method has been used in several pieces of research. (Schömig et al., 2020). In evaluating the decision-making framework for cloud computing (Opara-Martins et al., 2017), policy research discussed expert evaluation as a highly considered internal reliability and result-based legitimacy (Sager & Mavrot, 2021) and several other researchers. Determining the levels of verbal scales used to evaluate quantitative and qualitative parameters in expert evaluation is a serious issue when using expert assessment for decision-making evaluation (Litvak, 1996). However, to mitigate this issue, the five-point Likert scale was adopted to reduce the verbal or the use of technical terminology that may impact the evaluation and a also the questionnaire was based on a sample of the question used in (Opara-Martins et al., 2017) and they are reword and redesign to accommodate the research the article research questions.

Method

The expert evaluation utilized a quantitative survey to garner the expert view of the proposed decision support framework for the portability of cloud-native architecture. The survey was designed using a web-based questionnaire once used by (Opara-Martins et al., 2017) to evaluate a decision-making framework as a guide. The questionnaire comprised 26 questions developed through the guidelines prescribed in Shared Assessment Agreed upon Procedures and Standardized Information Gathering (SIG) Questionnaire. Participants are invited to interact with the framework through the questionnaire. However, the participants varied between IT professionals, managers, cloud architects, consultants, and developers with expertise in cloud computing, lower-middle-income, and cloud-native architecture.

Recruiting participants for expert evaluation was done by considering experts in the subject area to provide their feedback or evaluation of the framework. The target population is the potential participants for this kind of evaluation because they are more interested in the subject matter and are experts and ready to give such an evaluation (Zikmund et al., 2000). The participants were deliberately targeted as the respondents in this evaluation needed to understand the migration or portability in cloud computing with a focus on cloud-native architecture. They also needed to be familiar with lower-middle-income settings and decision-making. The online survey was designed to be clear to the experts. The data were systematically collected through the survey materials that enabled the researcher to download the result in a Microsoft Excel spreadsheet.

Result

The data collected from the survey were analyzed using the descriptive analysis method, where mean and standard deviation were estimated. The mean was used to show the central tendency of the data. In contrast, the standard deviation measures the index of the spread or variability in the data and helps understand the effects in the data. A higher mean value means greater importance or appropriateness towards the decision support framework, whereas a lower mean is not essential or appropriate and vice versa. Likewise, standard deviation for a set of values from the evaluation result reveals that these values are clustered closely about the mean. A significant standard deviation indicates the opposite where the participant in this is the expert analysis, followed by their feedback on the phases, steps, tasks and framework.

Descriptive analysis of the Framework Steps Important

Step Important	Variance	Mean	Standard Deviation
Step 1 – Determine the intention	0.63	4.15	0.79
Step 2 – Planning and information gathering	0.35	4.42	0.59
Step 3 – Choosing the deployment model	0.7	4	0.84
Step 4 – Vendor evaluation and selection	0.53	4.15	0.73
Step 5 – Contract and SLA	0.46	4.2	0.68
Step 6 - Execute the portability plan	0.44	4.37	0.67
Step 7 – Testing and documentation	0.74	4.4	0.86
Step 8 – Management and Optimization	0.63	4.35	0.79

Table 1: Response about the importance of the Steps

Finding how appropriate the step-by-step arrangement of the framework is important which is to show that the steps are appropriate for the organization to considered before making any decision to port into or through cloud-native architecture. Based on the expert it can be clearly observed that the responses are clear and mainly appropriate. **Table 1** provide a clear view on the expert consideration and how the data is spread through the descriptive statistics computation that show through the difference between average mean values for the steps is less that the least standard deviation (i.e., 0.59) for the step. This mean that the largest standard deviation from the sample mean (i.e., 0.86) is less than twice the smallest standard deviation (2*0.59=1.18) which further illustrates that the survey data is reasonably normal. After detail considering the output of the analyzed data it can be inferred that the steps are appropriate, and the evaluation results of the novel decision framework can be considered is reasonable.

Descriptive analysis of the Framework Steps Appropriate

To clarify that arrangement of the steps in the framework are appropriate based on the expert evaluation. However, it is an essential reason to confirm how important is the arrangement of the step. Majority of the expert which are the participant of the survey indicate that the step is important through the response that are mainly between very important to absolutely important. Consideration of the steps, individual steps such as Step 5 contact and SLA, Step 6 Execute the portability plan and Step 7 Testing and documentation indicate a form of interest from the participant. The interest can be clear seen through the data that show almost hundred percent between the range of very important and absolute important for how important the steps arrangement.

Table 2 consider the participant impact on the steps arrangement and the individual location of the step making using the same descriptive statistics computation to show the difference between average mean values for the steps is less that the least standard deviation (i.e., 0.50) for the step. This mean that the largest standard deviation from the sample mean (i.e., 0.77) is less than twice the smallest (2*0.50=1) which further illustrates that the survey data is reasonably normal. The output of the analyzed data can be concluded that the steps are important, and the evaluation results of the novel decision framework can be considered is appropriate.

Overall Effectiveness of the Framework

The author asks the participant about how effectively is the whole framework support organizations' decision-making process for migration to cloud-native architecture? The result from the framework depicts its effectiveness. As seen in Table 3, 45% of the respondents consider the framework an "extremely effective framework", and another 45% also consider this framework a "quite effective" one. From the result shown in the graph and table and the response by IT professionals, one can conclude the effectiveness of each task as opposed to its non-effectiveness.

Table 2: The appropriate the steps

Answer	**Response**	
Extremely effective (1)	45%	9
Quite effective (2)	45%	9
Moderately effective (3)	5%	1
Slightly effective (4)	5%	1
Not at all effective (5)	0%	0
Total	**100%**	**20**

5. INTEGRATING THE CLOUD NATIVE ARCHITECTURE PORTABILITY FRAMEWORK INTO ES

This section provides a clear critical view on how the cloud-native architecture portability framework was developed into an expert system after considering all the input of the human experts and other reasons discussed in Section 3. The expert system was developed using the web-based development tool ES-BUILDER (McGoo, 2013), and this shell is used for easy access via the Internet. This section critically discusses the different parts of the expert system and how the framework was integrated. The expert system consists of three significant parts Knowledge-based, Inference Engine and User interface, which are used in capturing knowledge (Liebowitz, 1995). The three components are explained in the subsequent sections.

Knowledge-Based

Knowledge could be defined as a set of facts and heuristics (rules of thumb) or the result of the process information; thus, the knowledge base must be complete, consistent and accurate (Liebowitz, 1995). The knowledge base contains knowledge from the human expert for a particular domain (Zhang et al., 2022). This part of the ES has a structure of rules in the form of IF condition THEN, also called the "Rule Base" (Ye & Wu, 2014). This is part of the expert system where the coding of the

Table 3: Effectiveness of the framework

Steps appropriate	Variance	Mean	Standard Deviation
Step 1 – Determine the intention	0.53	4.15	0.73
Step 2 – Planning and information gathering	0.43	4.32	0.65
Step 3 – Choosing the deployment model	0.6	4	0.77
Step 4 – Vendor evaluation and selection	0.53	4.25	0.73
Step 5 – Contract and SLA	0.21	4.3	0.46
Step 6 - Execute the portability plan	0.33	4.35	0.57
Step 7 – Testing and documentation	0.25	4.55	0.5
Step 8 – Management and Optimization	0.59	4.25	0.77

expert knowledge through the knowledge engineering system in which the structuring and presentation of knowledge are done with the programmable statement with IF, ELSE and THEN.

Inference Engine

The inference engine is the problem-processing part of the expert system. It controls the structure of the expert system that allows the expert to use search strategies to test different hypotheses to arrive at expert system conclusions (Rajabi et al., 2019). The inference engine for generating inferences over the knowledge base and this is done by applying the facts to the rules and determining the questions to be asked by the user in the user interface and in which order to ask them, and the output that needs to be presented to the user through the user interface. Expert systems conclusions drawn by the inference engine can be based on either deterministic or probabilistic knowledge. Two different inferencing methods are popular in ES: Forward Chaining and Backward Chaining.

User Interface

The user interface is part of the expert system where the user interacts with the system, effectively facilitating this interaction and communication (Liebowitz, 1995). Figure 3 shows the interaction between the user and the inference engine, which makes the expert system a solution for several domains such as medicine, civil and many others and valuable for this research. Because users can see the output of the decision after inputting or interacting with the system through the graphic user interface (GUI). An effective expert system with a quality user interface is necessary, and it improves the quality of the expert system.

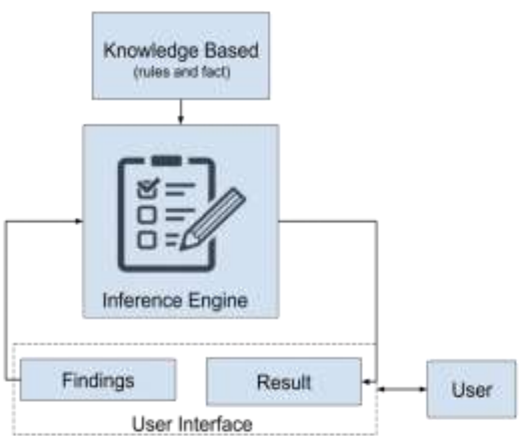

Figure 3: Basic Expert System Component (adapted from Rajabi et al., 2019).

5.1 Implementation of the Expert System

This section examines the implementation of the ES for the portability or migration in cloud-native architecture and how the author makes use of the ES BUILDER by McGoo to implement the Expert system.

1. The Shell

As mentioned above, the ES-Builder is the shell for the expert system. The ES-Builder is a web-based expert system shell within a built knowledge base with a decision tree modelling process for the development of the logic of the expert system. The program or shell was designed and built to assist expert system developers by giving them an interface to design and implement the expert system in an easy way to visualise, edit and present the systems to users. The ES-Builder is a type of expert system shell developed using a process of deductive reasoning (Matthew et al., 2016). In this case, the developer, the user, construct the expert system using a decision tree interface where attributes, values and conclusions are added as leaf nodes on the tree. Each node has a small integrated data set used to form the expert system's content when accessed online. After the developer finishes with the development of the expert system on the ES-Builder website, users can then have access to the expert system via the Internet. The following are steps that need to be taken to create an expert system in the ES-Builder expert system shell:

- Plan and Design the Expert System Efficiently using a clearly defined Universe of Discourse
- Identified all the conclusions and included them in the expert system
- Determined all the attributes that were tested by the expert system
- Identified all appropriate values for each attribute.
- Prepare extra notes on each possible attribute, value, and conclusion to further inform the user about the result of each search.
- Use suitable image graphics to illustrate each attribute, value, and conclusion during the search process if needed.

2. Decision Tree

The deductive logic of the ES is created through the Decision Tree View by entering the title details, attributes, values, and conclusions into a decision tree. Each step in the decision tree is called a node. A node that branches out of another node in the decision tree is called a branch node. A node may have branches to other nodes, and so on, until the decision tree is complete. There are some basic rules about how the tree can be formed and which branch nodes a particular type of node may accept. The nodes at the very ends of branches are called leaf nodes. Figure 4 is the diagrammatic view of the decision tree for the decision-supporting system for cloud-native portability ES based on the Decision-supporting framework.

3. The Knowledge Base

The knowledge base of the decision-supporting system for the cloud-native portability expert system sample was captured in Table 4 below, and the complete version can be found in

It shows the information stored in the ES in the form of facts and rules used in the coding or programming of the Expert System. The knowledge base is part of the expert system with the IF condition(s), AND Condition and THEN consequence's structure. This means IF statements are satisfied, the THEN will take place.

http://www.mcgoo.com.au/esbuilder/viewer/viewES.php?es=bf327779053a44343873ba2393fc6698

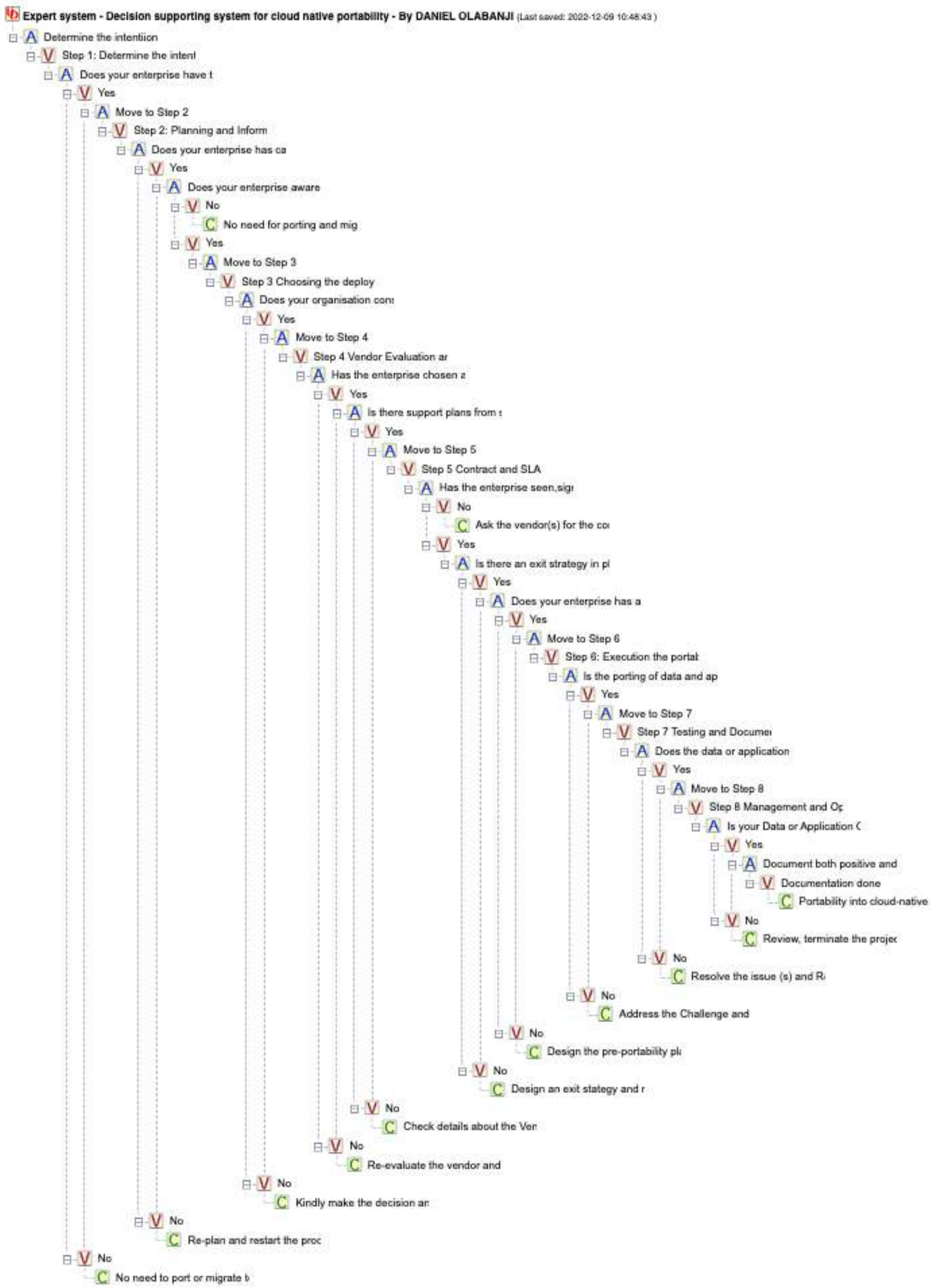
Expert system - Decision supporting system for cloud native portability - By DANIEL OLABANJI (Last saved: 2022-12-09 10:48:43)
A Determine the intentiion
V Step 1: Determine the intent
A Does your enterprise have t
V Yes
A Move to Step 2
V Step 2: Planning and Inform
A Does your enterprise has ca
V Yes
A Does your enterprise aware
V No
C No need for porting and mig
V Yes
A Move to Step 3
V Step 3 Choosing the deploy
A Does your organisation cons
V Yes
A Move to Step 4
V Step 4 Vendor Evaluation ar
A Has the enterprise chosen a
V Yes
A Is there support plans from
V Yes
A Move to Step 5
V Step 5 Contract and SLA
A Has the enterprise seen,sign
V No
C Ask the vendor(s) for the co
V Yes
A Is there an exit strategy in pl
V Yes
A Does your enterprise has a
V Yes
A Move to Step 6
V Step 6: Execution the portab
A Is the porting of data and ap
V Yes
A Move to Step 7
V Step 7 Testing and Documen
A Does the data or application
V Yes
A Move to Step 8
V Step 8 Management and Op
A Is your Data or Application
V Yes
A Document both positive and
V Documentation done
C Portability into cloud-native
V No
C Review, terminate the projec
V No
C Resolve the issue (s) and R
V No
C Address the Challenge and
V No
C Design the pre-portability pl
V No
C Design an exit stategy and r
V No
C Check details about the Ven
V No
C Re-evaluate the vendor and
V No
C Kindly make the decision an
V No
C Re-plan and restart the proc
V No
C No need to port or migrate t

Figure 4: Expert System: Decision supporting system for cloud-native portability decision tree

Table 4: *Decision supporting system for cloud-native portability expert system rule*

#	Rule
1	**IF** determine the intention to port or migrate into cloud-native architecture Step 1: Determine the intention. **AND** does your enterprise have the intention to port into cloud-native architecture? no **THEN** the Conclusion is **No need to port or migrate to Cloud-native architecture**.
2	**IF** determine the intention to port or migrate into cloud-native architecture Step 1: Determine the intention. **AND** does your enterprise have the intention to port into cloud-native architecture? yes **AND** move to Step 2. Step 2: Planning and Information Gathering **AND** does your enterprise has the capability to port or migrate to a cloud-native architecture? no **THEN** the Conclusion is **Do not port OR Re-plan and restart the process**.
3	**IF** determine the intention to port or migrate into cloud-native architecture Step 1: Determine the intention. **AND** does your enterprise have the intention to port into cloud-native architecture? yes **AND** move to Step 2. Step 2: Planning and Information gathering **AND** does your enterprise has the capability to port or migrate to a cloud-native architecture? yes **AND** does your enterprise aware of the policy, risks, tasks and processes no **THEN** the Conclusion is **No need for porting and migration**.
4	**IF** determine the intention to port or migrate into cloud-native architecture Step 1: Determine the intention. **AND** does your enterprise have the intention to port into cloud-native architecture? yes **AND** move to Step 2 Step 2: Planning and Information gathering **AND** does your enterprise has the capability to port or migrate to a cloud-native architecture? yes **AND** does your enterprise aware of the policy, risks, tasks and processes yes **AND** move to Step 3 Step 3: Choosing the Deployment model **AND** does your organisation consider the needed deployment model? no **THEN** the Conclusion is **Kindly make the decision and start all over again.**
5	**IF** determine the intention to port or migrate into cloud-native architecture Step 1: Determine the intention **AND** does your enterprise have the intention to port into cloud-native architecture? yes **AND** move to Step 2 Step 2: Planning and Information gathering **AND** does your enterprise has the capability to port or migrate to a cloud-native architecture? yes **AND** does your enterprise aware of the policy, risks, tasks and processes yes **AND** move to Step 3 Step 3: Choosing the Deployment model **AND** does your organisation consider the needed deployment model? yes **AND** move to Step 4 Step 4: Vendor Evaluation and Selection **AND** has the enterprise chosen a vendor(s) based on the needed criteria? no **THEN** the Conclusion is **Re-evaluate the vendor choice and select the right one for the project. Then the enterprise can start the decision support system again.**
6	**IF** determine the intention to port or migrate into cloud-native architecture Step 1: Determine the intention **AND** does your enterprise have the intention to port into cloud-native architecture? yes **AND** move to Step 2 Step 2: Planning and Information gathering **AND** does your enterprise has the capability to port or migrate to a cloud-native architecture? yes **AND** does your enterprise aware of the policy, risks, tasks and processes yes **AND** move to Step 3 Step 3: Choosing the Deployment model **AND** does your organisation consider the needed deployment model? yes **AND** move to Step 4 Step 4: Vendor Evaluation and Selection **AND** has the enterprise chosen a vendor(s) based on the needed criteria? yes **AND** is there support plans from the service provider on porting no **THEN** the Conclusion is **Check details about the Vendor and service provider** .

7	**IF** determine the intention to port or migrate into cloud-native architecture Step 1: Determine the intention **AND** does your enterprise have the intention to port into cloud-native architecture? yes **AND** move to Step 2 Step 2: Planning and Information gathering **AND** does your enterprise has the capability to port or migrate to a cloud-native architecture? yes **AND** does your enterprise aware of the policy, risks, tasks and processes yes **AND** move to Step 3 Step 3: Choosing the Deployment model **AND** does your organisation consider the needed deployment model? yes **AND** move to Step 4 Step 4: Vendor Evaluation and Selection **AND** has the enterprise chosen a vendor(s) based on the needed criteria? yes **AND** is there support plans from the service provider on porting yes **AND** move to Step 5 Step 5: Contract and SLA **AND** has the enterprise agreed and sign the contract and SLA? no **THEN** the Conclusion is **Ask the vendor(s) for the contract and SLA between the enterprise and the vendor. Then restart the expert system.**
8	**IF** determine the intention to port or migrate into cloud-native architecture Step 1: Determine the intention **AND** does your enterprise have the intention to port into cloud-native architecture? yes **AND** move to Step 2 Step 2: Planning and Information gathering **AND** does your enterprise has the capability to port or migrate to a cloud-native architecture? yes **AND** does your enterprise aware of the policy, risks, tasks and processes yes **AND** move to Step 3 Step 3: Choosing the Deployment model **AND** does your organisation consider the needed deployment model? yes **AND** move to Step 4 Step 4: Vendor Evaluation and Selection **AND** has the enterprise chosen a vendor(s) based on the needed criteria? yes **AND** is there support plans from the service provider on porting yes **AND** move to Step 5 Step 5: Contract and SLA **AND** has the enterprise agreed and sign the contract and SLA? yes **AND** is there an exit strategy or clause in the SLA? no **THEN** the Conclusion is **Design an exit strategy and restart** .
9	**IF** determine the intention to port or migrate into cloud-native architecture Step 1: Determine the intention **AND** does your enterprise have the intention to port into cloud-native architecture? yes **AND** move to Step 2 Step 2: Planning and Information gathering **AND** does your enterprise has the capability to port or migrate to a cloud-native architecture? yes **AND** does your enterprise aware of the policy, risks, tasks and processes yes **AND** move to Step 3 Step 3: Choosing the Deployment model **AND** does your organisation consider the needed deployment model? yes **AND** move to Step 4 Step 4: Vendor Evaluation and Selection **AND** has the enterprise chosen a vendor(s) based on the needed criteria? yes **AND** is there support plans from the service provider on porting yes **AND** move to Step 5 Step 5: Contract and SLA **AND** has the enterprise agreed and sign the contract and SLA? yes **AND** is there an exit strategy or clause in the SLA? yes **AND** does your enterprise has a pre-portability plan? no **THEN** the Conclusion is **Design the pre-portability plan**.
10	**IF** determine the intention to port or migrate into cloud-native architecture Step 1: Determine the intention **AND** does your enterprise have the intention to port into cloud-native architecture? yes **AND** move to Step 2 Step 2: Planning and Information gathering **AND** does your enterprise has the capability to port or migrate to a cloud-native architecture? yes **AND** does your enterprise aware of the policy, risks, tasks and processes yes **AND** move to Step 3 Step 3: Choosing the Deployment model **AND** does your organisation consider the needed deployment model? yes **AND** move to Step 4 Step 4: Vendor Evaluation and Selection **AND** has the enterprise chosen a vendor(s) based on the needed criteria? yes **AND** is there support plans from the service provider on porting yes **AND** move to Step 5 Step 5: Contract and SLA **AND** has the enterprise agreed and sign the contract and SLA? yes **AND** is there an exit strategy or clause in the SLA? yes

	AND does your enterprise has a pre-portability plan? yes **AND** move to Step 6 Step 6: Execution the portability **AND** is the porting of data and application successful? no **THEN** the Conclusion is **Address the Challenge and Reconfigure**.
11	**IF** determine the intention to port or migrate into cloud-native architecture Step 1: Determine the intention **AND** does your enterprise have the intention to port into cloud-native architecture? yes **AND** move to Step 2 Step 2: Planning and Information gathering **AND** does your enterprise has the capability to port or migrate to a cloud-native architecture? yes **AND** does your enterprise aware of the policy, risks, tasks and processes yes **AND** move to Step 3 Step 3: Choosing the Deployment model **AND** does your organisation consider the needed deployment model? yes **AND** move to Step 4 Step 4: Vendor Evaluation and Selection **AND** has the enterprise chosen a vendor(s) based on the needed criteria? yes **AND** is there support plans from the service provider on porting yes **AND** move to Step 5 Step 5: Contract and SLA **AND** has the enterprise agreed and sign the contract and SLA? yes **AND** is there an exit strategy or clause in the SLA? yes **AND** does your enterprise has a pre-portability plan? yes **AND** move to Step 6 Step 6: Execution the portability **AND** is the porting of data and application successful? yes **AND** move to Step 7 Step 7: Testing and Documentation **AND** does the data or application testing successful? no **THEN** the Conclusion is **Resolve the issue (s) and Re-test**.
12	**IF** determine the intention to port or migrate into cloud-native architecture Step 1: Determine the intention. **AND** does your enterprise have the intention to port into cloud-native architecture? yes **AND** move to Step 2 Step 2: Planning and Information gathering **AND** does your enterprise has the capability to port or migrate to a cloud-native architecture? yes **AND** does your enterprise aware of the policy, risks, tasks and processes yes **AND** move to Step 3 Step 3: Choosing the Deployment model **AND** does your organisation consider the needed deployment model? yes **AND** move to Step 4 Step 4: Vendor Evaluation and Selection **AND** has the enterprise chosen a vendor(s) based on the needed criteria? yes **AND** is there support plans from the service provider on porting yes **AND** move to Step 5 Step 5: Contract and SLA **AND** has the enterprise agreed and sign the contract and SLA? yes **AND** is there an exit strategy or clause in the SLA? yes **AND** does your enterprise has a pre-portability plan? yes **AND** move to Step 6 Step 6: Execution the portability **AND** is the porting of data and application successful? yes **AND** move to Step 7 Step 7: Testing and Documentation **AND** does the data or application testing successful? yes **AND** move to Step 7 Step 8 Management and Optimisation **AND** is your Data or Application Operating Successfully no **THEN** the Conclusion is **Review and Terminate the project and Document the failure**.
13	**IF** determine the intention to port or migrate into cloud-native architecture Step 1: Determine the intention **AND** does your enterprise have the intention to port into cloud-native architecture? yes **AND** move to Step 2 Step 2: Planning and Information gathering **AND** does your enterprise has the capability to port or migrate to a cloud-native architecture? yes **AND** does your enterprise aware of the policy, risks, tasks and processes yes **AND** move to Step 3 Step 3: Choosing the Deployment model **AND** does your organisation consider the needed deployment model? yes **AND** move to Step 4 Step 4: Vendor Evaluation and Selection **AND** has the enterprise chosen a vendor(s) based on the needed criteria? yes **AND** is there support plans from the service provider on porting yes **AND** move to Step 5 Step 5: Contract and SLA **AND** has the enterprise agreed and sign the contract and SLA? yes

	AND is there an exit strategy or clause in the SLA? yes **AND** does your enterprise has a pre-portability plan? yes **AND** move to Step 6 Step 6: Execution the portability **AND** is the porting of data and application successful? yes **AND** move to Step 7 Step 7: Testing and Documentation **AND** does the data or application testing successful? yes **AND** move to Step 7 Step 8 Management and Optimisation **AND** is your Data or Application Operating Successfully yes **AND** document both positive and negative experience and processes documentation Successful **THEN** the Conclusion is **Portability into cloud-native architecture is successful.**

can

CONCLUSION

After a critical discussion on the need for a decision support framework for portability into or through cloud-native architecture and the consideration of the existing frameworks, there is a missing gap that this paper is aiming to fill. This paper presents a novel cloud-native portability framework and the stages of modification that a cloud-native architecture adoption should undergo before deciding to migrate, what to do during migration, and after the migration to a cloud-native architecture. All these considerations and feedback from human experts through surveys and questionnaires led to the designing and development of the framework. The framework is statistically evaluated through a simple descriptive statistical method which brings some changes in the original steps. The final framework is then incorporated into an Expert System (ES) to validate the framework which is to check if the decision-supporting system can function automatically and provide interaction between the decision-makers and the framework, which is done to provide the user with a functioning view of the framework.

The expert system was developed using an ES-BUIDER web-based development tool for easy access and usability capability of the framework. The paper presents all the stages involved in ES development and how to use the expert system to validate the framework. The output of the designed and validation expert system show that the decision-supporting system or framework can be the solution that can be used to support organizational decision-making when considering the adoption of cloud-native architecture. In addition, the expert system also provides a 13 technical consideration for the user in which at the end of the user or organization follow for rightful decision. Validating the steps in the framework worked perfectly in the expert system which shows that the decision process in the framework work using artificial intelligence such as expert system.

Future work

This research provides an expert system's approach in validating cloud-native portability or migration framework. However, further research is needed to implement the framework into a more advanced artificial intelligence, which could provide a knowledge-based decision-supporting system for the cloud-native architecture portability and migration in organizations.

http://www.mcgoo.com.au/esbuilder/viewer/viewES.php?es=bf327779053a44343873ba2393fc6698

REFERENCES

Andrikopoulos, V., Binz, T., Leymann, F., & Strauch, S. (2013). How to adapt applications for the Cloud environment: Challenges and solutions in migrating applications to the Cloud. *Computing*, *95*(6), Article 6. https://doi.org/10.1007/s00607-012-0248-2

Chung, K., Boutaba, R., & Hariri, S. (2016). Knowledge based decision support system. *Information Technology and Management*, *17*(1), Article 1.

CNCF. (2019). *Cloud-native Definition v1.0*. https://github.com/cncf/toc/blob/master/DEFINITION.md

Feigenbaum, E., & Bond, A. (1981). State of the Art Report on Machine Intelligence. In *Expert Systems in the 80's*. Pergamon-Infotech.

Kratzke, N., & Quint, P.-C. (2017). Understanding cloud-native applications after 10 years of cloud computing-a systematic mapping study. *Journal of Systems and Software*, *126*, 1–16.

Liebowitz, J. (1995). Expert systems: A short introduction. *Engineering Fracture Mechanics*, *50*(5–6), Article 5–6.

Litvak, B. (1996). Expert assessments and decision making. *M.: Patent*, 298.

Liu, F., Tong, J., Mao, J., Bohn, R., Messina, J., Badger, L., & Leaf, D. (2011). NIST cloud computing reference architecture. *NIST Special Publication*, *500*(2011), 1–28.

Matthew, O., Buckley, K., Garvey, M., & Moreton, R. (2016). Multi-tenant database framework validation and implementation into an expert system. *International Journal of Advanced Studies in Computers, Science and Engineering*, *5*(8), Article 8.

Opara-Martins, J. (2017). *A decision framework to mitigate vendor lock-in risks in cloud (SaaS category) migration.*

Opara-Martins, J., Sahandi, M., & Tian, F. (2017). A holistic decision framework to avoid vendor lock-in for cloud saas migration. *Computer and Information Science*, *10*(3).

Poleshchuk, O. M. (2018). *Creation of linguistic scales for expert evaluation of parameters of complex objects based on semantic scopes*. 1–6.

Rajabi, M., Hossani, S., & Dehghani, F. (2019). A literature review on current approaches and applications of fuzzy expert systems. *ArXiv Preprint ArXiv:1909.08794*.

Sager, F., & Mavrot, C. (2021). Participatory vs expert evaluation styles. In *The Routledge Handbook of Policy Styles* (pp. 395–407). Routledge.

Samreen, F., Blair, G. S., & Elkhatib, Y. (2020). Transferable knowledge for low-cost decision making in cloud environments. *IEEE Transactions on Cloud Computing*, *10*(3), Article 3.

Schömig, N., Wiedemann, K., Hergeth, S., Forster, Y., Muttart, J., Eriksson, A., Mitropoulos-Rundus, D., Grove, K., Krems, J., & Keinath, A. (2020). Checklist for expert evaluation of HMIs of automated vehicles—Discussions on its value and adaptions of the method within an expert workshop. *Information*, *11*(4), 233.

Toffetti, G., Brunner, S., Blöchlinger, M., Spillner, J., & Bohnert, T. M. (2017). Self-managing cloud-native applications: Design, implementation, and experience. *Future Generation Computer Systems*, *72*, 165–179. https://doi.org/10.1016/j.future.2016.09.002

Zhang, Y., Chen, Y., & Li, X. (2022). Integrated Framework of Knowledge-Based Decision Support System for User-Centered Residential Design. *Expert Systems with Applications*, 119412.

Zikmund, W. G., Babin, B., Carr, J., & Griffin, M. (2000). Business research methods 6th edition. *Fort Worth, Texas*.