2018

An Analysis of Software Testing Practices on Migrations From on Premise to Cloud Hosted Environments

Ronan Mullen

Technological University Dublin

An Analysis of Software Testing Practices on Migrations from On Premise to Cloud Hosted Environments

Ronan Mullen

Dublin Institute of Technology, c02723158@mydit.ie

A dissertation submitted in partial fulfilment of the requirements of
Dublin Institute of Technology for the degree of
M.Sc. in Computing (Advanced Software Development)

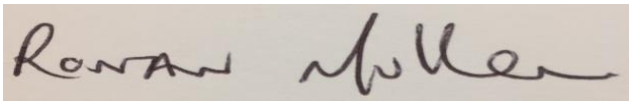
2018

DECLARATION

I certify that this dissertation which I now submit for examination for the award of MSc in Computing (Advanced Software Development), is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the test of my work.

This dissertation was prepared according to the regulations for postgraduate study of the Dublin Institute of Technology and has not been submitted in whole or part for an award in any other Institute or University.

The work reported on in this dissertation conforms to the principles and requirements of the Institute's guidelines for ethics in research.

Signed: 

Date: **15th June 2018**

ABSTRACT

This research project examines the differences between software testing practices that are carried out on software that is installed locally (i.e. on premise) versus software that has migrated to a cloud hosted environment. In conjunction with this, focus was placed on determining what methodologies and frameworks are in existence for assisting with software migrations to the cloud. The reason for carrying out this research project was that the transition to cloud computing is becoming more and more mainstream, as a result organisations are required to focus their efforts on how best to move their software to the cloud while ensuring that its functionality remains intact.

To determine what software testing differences are apparent and what migration strategies are available, existing literature in this area was reviewed and qualitative data was collected. Once all data was collated, results showed that there were in fact key differences in the complexity of software testing that was required for software that has migrated to the cloud. In addition, no generic strategy was found for carrying out a software migration to the cloud.

Based on the findings of this research project, it is paramount that an organisation which plans on migrating any of if its software or infrastructure to the cloud needs to have a clear road map in place prior to any development or migration work commencing. By clearly planning and executing this process, clear testing and validation phases can be put in place which will ensure than an element of precaution and prior validation can be incorporated into the transition to the cloud.

Keywords: Cloud, Migrations, Software, testing, SaaS, PaaS, IaaS.

ACKNOWLEDGEMENTS

I would like to thank my supervisor Diana Carvalho E Ferreira for all of her guidance, reassurance and enthusiasm throughout this dissertation. I would also like to thank and my friends and family for their constant support.

Contents

List of Figures

List of Acronyms

SME – Small and Medium Sized Enterprise

AWS - Amazon Web Services

IoT - Internet of Things

SDLC - Software Development Lifecycle

CSP – Cloud Service Provider

IT - Information Technology

SaaS – Software as a Service

PaaS – Platform as a Service

IaaS – Infrastructure as a Service

AI - Artificial Intelligence

QA - Quality Assurance

OS - Operating System

SLA - Service Level Agreements

GCC - GNU Compiler Collection

JDK - Java Development Kit

CRM - Customer Relationship Management

EC2 - Amazon Elastic Compute Cloud

VB - Visual Basic

MVP - Minimum Viable Product

URL - Uniform Resource Locator

GUI - Graphical user interface

QTP - Quick Test Professional

UFT - Unified Functional Testing

POC - Proof of Concept

List of Appendices

CHAPTER 1: INTRODUCTION

1.1 Background

In a modern business environment the concept of an SME creating a budget for its computing infrastructure (servers, databases, hardware etc.) is becoming less and less common. There has been a huge shift towards cloud hosted solutions which allow companies to rent all the services it needs rather than having to invest heavily in this costly equipment upfront (Carcary *et al.*, 2014).

This change has been facilitated by increased accessibility to these types of cloud hosting options through the likes of AWS and Microsoft Azure. These solutions allow a company or start-up of any size to rent the services that they may require such as; storage, data analytics or IoT resources. In addition to accessing these services, users can increase or decrease their resource allocation as required resulting in potentially limitless system scalability.

Cloud computing not only brings new business opportunities, but also causes some major impacts on software testing and maintenance (Gao *et al.*, 2013). This change in approach means that companies are now moving existing software installations (on premise) to cloud hosted environments with a view to saving money and allowing for straightforward system scalability.

1.2 Project Objectives

The aim of this research project is to understand if software testing practices have evolved to meet the needs of software which has migrated from an on premise installation to a cloud hosted environment.

Although hosting and migrating software to a cloud is not a new concept, its popularity and uptake has grown exponentially over the past number of years. This shift has resulted in changes to the way that software is developed and as a result tested. Organisations have also had to come up with frameworks and methodologies to carry

out this migration process based on recommendations, software structure and past experiences.

A software installation which has been moved from one environment to another should go through a rigorous testing cycle to determine if any issues have arisen. But this type of migration has not been factored into the original SDLC and therefore it is not clear if all of the existing software testing phases which were previously carried out on a routine basis are fit for purpose for a cloud hosted environment.

Software testing is a key aspect of the software development lifecycle. Previously, software testing focused on software programs which had been installed on "local" computers i.e. on premise. Since the industry trend has shifted significantly in favour of cloud hosted software, software testing tools, techniques and migration strategies have had to evolve and update in line with these changes.

1.3 Project Challenges

The challenges aligned with this project are mainly in relation to gathering data that is current, relevant and of interest for this topic. At the outset of this project, there was a lot of information available which was released by CSP`s, all of which focused on their specific products and all of the benefits associated with a move to their offering. This type of information may not be of use as there could be a level of bias which could feature in the associated material. As a result, it will be difficult to locate informative and impartial information.

As part of this study, the aim is to identify and interview individuals who have hands on industry experience of developing, testing and migrating software to the cloud. A challenge will be locating the relevant individuals and gaining their potentially useful and meaningful insight into this area.

CHAPTER 2: BACKGROUND RESEARCH

2.1 Cloud Computing

Cloud Computing as a concept has been in existence for over twenty years and has been adopted into the mainstream thanks to the likes of Amazon AWS and Microsoft`s Azure cloud offerings. At a very high level Cloud Computing enables users to switch from investing in costly IT equipment (servers, databases etc.) to renting these services from on-demand "virtual" environments i.e. cloud hosting.

Cloud Computing is internet based and works by leveraging powerful remote servers to enable the sharing of data-processing tasks, centralized data storage, and online access to computer services or resources. Cloud Computing allows for easy scalability and access to powerful databases, analytics, security and software functionality.

Cloud Computing resources are deployed in three main forms namely; public cloud, private cloud and hybrid cloud, each with their own specific purpose. In a public cloud scenario all resources are provided by a third party provider (such as AWS or Azure). Functionality such as virtual machines, data storage or payments modules can be leveraged over the internet by users on a free or paid usage model.

A private cloud is an implementation solely for one organisation with resources and performance dedicated to them. A hybrid cloud deployment is a combination of public and private cloud resources as well as traditional IT equipment all of which merge together to meet the needs of certain business requirements (Zhang, 2018).

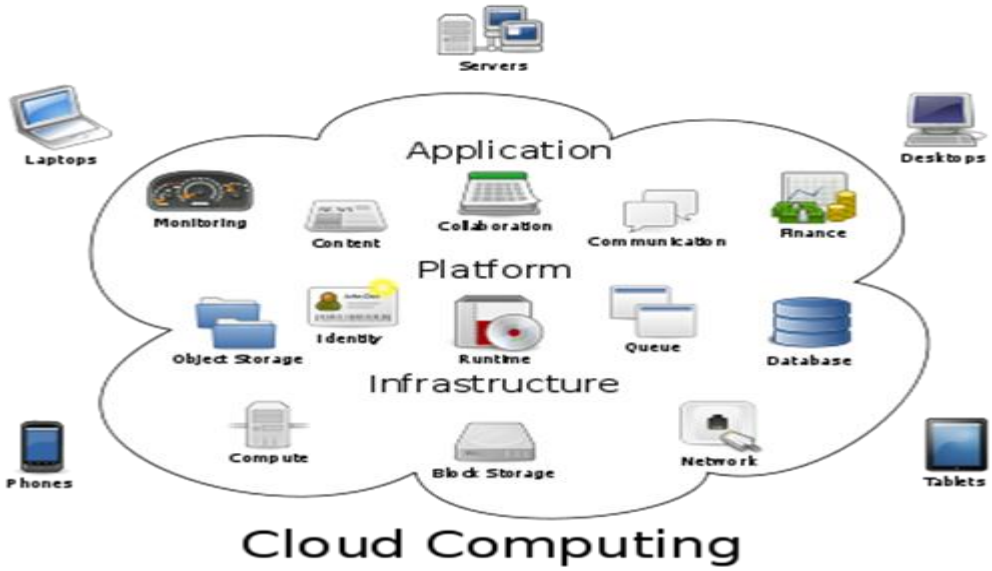


Figure 1. Cloud Computing Overview ("Introduction about Cloud Computing - Salesforce," 2013)

With the introduction of cloud computing, the concept of "virtualisation" has become more and more prevalent. Virtualisation is the ability to virtualise something such as an operating system or database. This means that rather than having a physical server for example, a user can create a virtual version and enable it on the cloud. This capability allows for resources to be pooled and shared between multiple possibly dispersed or unaware users. In practice, this allows for reductions in hardware costs, workload or resource optimisation and enables a large amount of flexibility, Salesforce (2013). This concept allows also for economies of scale to be introduced for cloud providers who can leverage this capability and provided cloud hosting services on a much larger, more cost effective scale.

One of the major concepts in cloud computing is that of multi-tenancy. Multi-tenancy is the ability for a single instance of a software application to be split out and used concurrently by multiple users, each of which are called tenants. This allows multiple tenants to access the same software package for example and customise its functionality to suit their needs while still being standalone and with no data being shared with other concurrent users (Ashalatha & Agarkhed, 2016).

There are many benefits to multi-tenancy such as shared infrastructures, constant improvements to software and cost savings to both users and providers. For service

providers multi-tenancy allows for reduced costs in terms of maintenance and updates. There is one codebase that serves multiple customers, this reduces the amount of updates that need to be made if a change or system updated is required, Whatis.com (2014).

Containers are a way of releasing software in a reliable standalone manner which reduces the risk of issues being introduced. The container can include all of the information needed for the software to function as expected (code, libraries, settings). This allows developers to move their applications between multiple environments and isolate or sandbox it as required to ensure that it is not negatively impacted by external inputs or dependencies. This process reduces resource requirements because the software is self-contained and as a result only the required computational power is needed (Barik *et al*., 2016). An example of a popular container tool is Docker[1] which combines and packages software into standalone modules which can updated, deployed and reused as required.

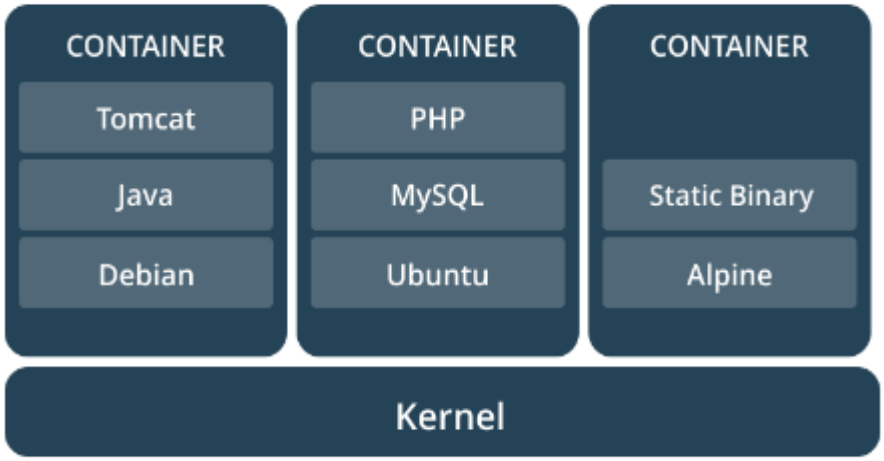


Figure 2. Containerised software example, Docker (2017)

The term cloud computing is a very broad concept which describes every possible implementation and flavour of this service. At a more granular level there are three different cloud computing models each consisting of unique aspects and functionality which suits different business and functionality scenarios. These are SaaS, PaaS, and IaaS.

[1] https://www.docker.com/what-container

2.2 Software as a Service (SaaS)

The SaaS cloud model enables third party providers to host software applications and make them available over the internet to customers who can leverage them in a cost effective manner. Customers can then log in from any computer in the world and use these applications in a straightforward manner and with limited integration work required. Some popular and successful SaaS providers are Salesforce, Dropbox and Microsoft Office 365. These applications are either free or work on a paid subscription basis.

The main benefit of this model for customers and users is that do not need to develop and maintain their own software applications, they can rely on third party providers to provide this service, perform updates and maintenance systems in general. These factors reduce the reliance on the customer's time and overheads. However, this option is not suitable for all users as the offerings provided by the SaaS model may be too generic and not fit the specific requirements that users may need. As a result, a more bespoke system may be required (Singh *et al*., 2016).

2.3 Platform as a Service (PaaS)

The PaaS cloud computing model allows users to access a cloud provider's application and leverage preconfigured software tools (databases, security applications etc.) with little or no set up/configuration required. This allows users to rapidly create their own cloud applications using provider specific tools, functionality and pre built applications.

One of the main benefits of this model is that is extremely cost effective. The user can purchase the use of a specific software tool for a limited time rather than having to buy this software, install it and then maintain it. This in effect allows for a pay as you go model which is very attractive to users with ever changing needs. This also allows for huge scalability and is one of the key benefits of cloud computing in general (Singh, Sharma *et al*., 2016). However, there are a certain amount of restrictions aligned with PaaS because the user may become locked in to the provider's choice of language and software tool selection which may be an unnecessary risk for some organisations.

2.4 Infrastructure as a Service (IaaS)

This cloud computing model allows users to run any application they require on cloud hardware and configurations of their choice. It enables companies to migrate their existing on premise applications or data for example to a cloud hosted service with a view to reducing IT overheads, costs and possibly increasing the performance and scalability of existing systems.

Depending on the budget of the organisation there are a number of options available in terms of IaaS. They can rent physical servers or virtualised servers both of which can be provided privately or on demand as required from public cloud providers. It is also possible to have a mix of both which is the hybrid option.

In terms of service providers offering cloud hosting solutions, Forbes (2017) highlights that there are a number of companies who hold a significant share in the market worldwide such as; Microsoft, Amazon, IBM, Salesforce and SAP. This market was originally a race to see who could build the biggest and best service offerings for customers. Now that all services are performant and in everyday use by customers, companies are aiming to attract customers via bespoke add-ons such as AI, machine learning and security features.

2.5 External dependencies on cloud offerings

One of the important differences between hosting software on the cloud and hosting it locally, on an internal server for example is that once it is migrated to the cloud the management and configuration of it may longer be under an organisations control. This is an important consideration when choosing the type of implementation (IaaS, SaaS or PaaS) that software will be migrated to.

In order to ensure that dependencies between the cloud provider and the user are fully understood, there are a number of design considerations that need to be made when architecting and choosing a new cloud hosting option such as; maintenance structures, compliance and QA procedures (Ladhe *et al.,* 2015).

CHAPTER 3: LITERATURE REVIEW

3.1 Introduction

The aim of this chapter is to document all of the research that was carried out at the outset of this project. This research was carried out with a view to understanding the knowledge that was available in the area of software testing practices as applied to software that has been migrated from on premise locations to cloud hosted environments.

In addition to the above, this research aimed to determine if in fact there were any existing standardised methodologies or frameworks available to assist users and organisations in completing these types of software migrations.

The outcome of this research will assist in identifying if there are any differences between the testing practices carried out on software that has been migrated locally i.e. internal server – internal server or to the cloud i.e. internal server – cloud hosted environment and if any standardised approaches are available. By heavily researching this topic, it is envisioned that gaps in research can be identified which will highlight the areas in which further investigation is required.

3.2 SaaS –vs- PaaS– vs – IaaS

One of the main considerations when migrating locally installed software to cloud hosted environments is determining what implementation is the most suitable for the specific project needs. There are many variations of cloud offerings, each of which may suit different project requirements. Depending on a user's preference, they may choose a certain implementation for a certain project or application (Dhanasekaran & Vasudevan, 2017).

In terms of software testing, the environment in which an application is hosted is extremely important. The more control a user has of the environment, the clearer visibility that they have about the types of inputs and dependencies that their

application may be under. In a SaaS style implementation for example, the cloud vendor has full control on all aspects of the software and services that are provided to the user. This may not be an issue if all the user requires is a “plug and play” style implementation such as an email or payments tool for example.

For many companies, a SaaS style cloud solution will not be a feasible option. This is because if a company wants to migrate something (e.g. a software application) to the cloud, SaaS will be useless. This is because in SaaS all applications and data are managed by the cloud provider. The only suitable options for this type of purpose would be IaaS or PaaS. With the above point and software testing in general in mind, focus needs to be placed on the types of dependencies that are put on an application that has been migrated to a SaaS or PaaS style cloud environment.

In terms of IaaS, companies have full control over the software application, data, runtime, middleware and operating system. Because of this companies can manage, schedule and control updates to each of these components and as a result then have full visibility and control of the types of changes that are being rolled out. This allows for testing cycles to be planned and sequenced in a manner which takes into account the changes that have been made to any part of the environment. This also means that, in this type of implementation, companies will have increased overheads and management responsibilities compared to a PaaS style configuration.

However, with a PaaS type cloud implementation, companies are at the mercy of cloud providers when it comes to upgrades to runtime, middleware and operating systems or components. If a company has their application hosted on a PaaS type cloud then they may be subject to mandatory updates and rollouts of new releases. These types of updates may introduce issues or software bugs to the host application and major unexpected problems may be encountered.

In relation to the software testing cycles carried out on cloud hosted applications, mandatory upgrades and new releases may be extremely difficult to account for. An upgrade or change released by the cloud provider will need to be verified by the QA team to ensure that no vulnerabilities or issues have been introduced to the host

application. This may be difficult to do if there is limited information available in terms of the actual changes that have been made to all connected systems.

This is an important concept to bear in mind when deciding what type of cloud solution is best for each particular project. A paper released by Maenhaut *et al*., states that when software is hosted on an IaaS style implementation then the user has full control and access to the OS, middleware, and runtime. However, with a PaaS offering, the customer only has control of the application and data which "brings some limitations, such as the selected OS and supported frameworks and libraries". This suggests that there are a number of factors which are out of the control of the user and dependant on the cloud provider.

This paper also states that when software or application updates are required a second instance of the environment can be spun up with a view to testing and verifying that no issues have been introduced. This can be done and planned for in conjunction with the device manager (i.e. cloud administrator) and the administration service (i.e. software release manager), according to this paper. However, there is no mention of a process in place for dealing with or the testing of unplanned or unscheduled updates to the system (OS), middleware or runtime configurations. This may be one of the limitations or pain points when it comes to choosing a PaaS style solution.

Due to the increasing popularity of cloud computing more and more systems and tools are being integrated into the cloud. As a result of this increase, the speed of change that is required in the industry is growing rapidly. The concept of continuous system integration has become increasingly prevalent in all aspects of the IT ecosystem. CSP`s have embraced this idea with a view to constantly upgrading and tweaking all aspects of their offerings.

A software testing book released by Pol & Mengerink, (2013) has highlighted this new trend as a potential issue when it comes to software that is hosted on cloud environments. CSP`s are constantly doing releases and updates to all areas of their product offerings with patches and changes often being pushed out on a daily basis, some of which are unannounced and inadequately documented.

As a result of this, this book states that there is now a need for a “permanent regressive system integration test” (Pol & Mengerink, 2013) to ensure that there is cohesion and full functionality within the host application. Also stated is that this should be a major factor when choosing the type (SaaS, IaaS or PaaS) of cloud offering and also the CSP to choose because continuity and clarity are key when it comes to system upgrades in cloud computing environments.

Business continuity between CSP`s and cloud computing users is documented as an important factor regardless of whether a SaaS, PaaS or IaaS implementation is chosen. Based on research for this project, it has been noted that it is essential that QA engineers and managers have visibility of the changes that are made in the environments in which they are working with and testing.

In some cases, QA engineers have logged into systems only to discover that upgrades have been made overnight and unexpected changes have been released (“Migrating Applications to Public Cloud Services: Roadmap for Success,” 2018) . With this in mind, it is essential that all teams involved in the verification and quality of software are proactive when it comes to upgrades made on cloud hosted environments. The most straightforward way of ensuring this is through clarity and communication i.e. SLAs with CSP`s.

3.3 Internal software migration (Server – Server)

It is important to note that as part of the research carried out for this project, limited information was available in the area of internal software migrations. This type of migration refers to a scenario whereby software is moved locally from one internally managed server to another. This may happen for a number of reasons such as server retirement, performance upgrades or change of hardware provider.

Compared to the amount of literature available in relation to software migrations to the cloud, internal migration strategies and processes are not well documented. In the limited information available, this type of migration mainly focuses on shifting capacity between servers and performance optimisation techniques (Canalir *et al.*,

2018). As a result, software testing methodologies, migration frameworks and software suitability are not discussed in detail.

3.4 Software suitability for cloud migration

One of the most important decisions to make during a cloud migration project is whether or not the software which is to be migrated to the cloud is suitable to be moved and if it will be robust and usable one migrated.

As discussed by Parveen & Tilley, (2010) it is extremely important to identify when is the most suitable time to migrate software applications to cloud hosted environments. In order to understand this further an approach was taken to investigate this from two different angles, a) the testing cycles that are carried out on the application b) the components of the application which is being tested.

As part of this research, there is a focus on software testing and its role and overheads within an organisation as well as the high amount of technical detail that needs to be included in a large scale software migration project. The purpose of this body of work aims to show the reader that each time a testing cycle is carried out a system may need to be restored, reinitiated or reconfigured. All of this type of work may take a lot of time and added complexity could be added to the cycle if an application is migrated to the cloud.

In order to demonstrate the issues highlighted above, two experiments were carried out using various Junit test cases which were set up to test Hadoop and Spring frameworks as well as test cases for a GCC non-Java application, both of which were migrated to a cloud hosted environment. Both of the above approaches had a number of existing dependencies which varied in line with the complexity and installation requirements of the application which was being tested.

The results from the above experiments varied depending on the application that was being migrated. In terms of the Junit test cases, the migration to the cloud hosted environment displayed very positive results. Prior to this migration, the execution of large volumes of test cases on a developer's machine took a number of hours. Once

migrated to the cloud, the length of time that this process took was reduced significantly. The main reason for this outcome was that the only dependency on JUnit test cases is that JDK is installed on the host environment. As a result, there is little room for errors and configuration issues to be introduced.

In the case of migrating the GCC application to the cloud the results were not as positive. This application is tightly coupled and complex in terms of the environment that it needs to be installed on. For these reasons it transpired that there was a high amount of development work required to migrate this application to the cloud and then ensure its functionality. The amount of work required outweighed the benefits gained by being cloud hosted.

Analysis was carried out as part of this research in relation to the software testing cycles that benefit most from being migrated to the cloud. The findings showed that in the majority of circumstances and once the tests have been correctly written that unit testing can benefit significantly from being cloud hosted. The results were also positive in relation to automation and performance testing. Once the volume is large enough to justify this transition, the benefits and resource savings can be substantial.

As a result of carrying out the above research it has been determined that due to the significant shift in favour of cloud computing, software testing cycles and practices have had to evolve on a constant basis. This is because the hosting environments and software itself has changed significantly, this has led to a requirement for software testing components to update to meet this changing trend. Experimentation and research have shown that in general, migrating software to the cloud can have many benefits if the software itself is suitable

3.5 Legacy applications and their move to the cloud

As noted previously and in research documented on this topic (Rowe *et al.*, 2013), there are significant gains in terms of computing power that can be associated with cloud computing. However, this comes with a caveat because when the option of migrating to the cloud is being assessed many software architects and organisations do

not factor in the costs associated with the actual migration process as well as the suitability for the application to be hosted on the cloud.

In order to understand and investigate the above point, research has been carried out to document the migration of a text mining application from a local installation (on premise) to a cloud hosted environment, Rowe (2013). The aim of carrying out this complicated experiment was to formulate a methodology that can potentially be followed by software architects who need to evaluate the suitability of moving portions of their business i.e. software infrastructure to the cloud.

The experiments documented in this research are carried out with a view to being from a software architects perspective. They aim to evaluate the pros and cons of moving a legacy application to the cloud and the suggested pathway that should be followed. The limitations of the current application are related to storage and the handling and processing of large amounts of data.

An iterative process was adopted for the migration of the text mining application to the cloud. This began with benchmarking tests being carried out against the application to determine the execution times achieved in various scenarios. This was essential so that as the development and migration process continued, the performance of the application could be mapped. In order to fully test the application, each of the benchmark tests used datasets of 4 different sizes, each of which were ran 3 times (12 tests in total).

In total the migration process took 6 separate iterations, each of which were tested using the methods outlined above. Testing showed that once the migration process was complete, a performance increase of approx. 30% was noted on the cloud hosted environment. By following an iterative process, it was possible to map this performance increase steadily as the application was migrated.

The methodologies followed in this migration process were broken down into three main sections. Standalone mode was carried out over iterations 1, 2 & 3 and consisted of setting up that basic Java application and Hadoop framework on the cloud with a

view to validating that the process was feasible. Once complete, it was possible to establish a test bed and therefore benchmark the application performance. This was essential for testing going forward as there was now a bar which could be tested against.

Iteration 4 was referred to as Pseudo-Distribution mode and consisted of a number of experiments which were carried out to compare the performance of the Hadoop applications in the new cloud environment. As part of this iteration, learnings were gained in terms of the configurations required to enable the new environment installation and the configuring of Hadoop in two clustered nodes. Once functioning, it was possible to compare this updated instance with the iterations implemented in the Standalone mode and draw on the possible benefits of having multiple Hadoop processes.

Experiments carried out in iteration 5 & 6 were grouped and referred to as Distribution Mode. The aim of this body of work was to test the scalability potential of the newly migrated Hadoop environment. To carry out these experiments, the Hadoop environment was mirrored over two/three clustered nodes and all ran in unison. It was then possible to run tests to show if there were increases/decreases in performance on the application and as a result, prove potential future scalability.

As highlighted previously, the aim of this body of work was to determine the feasibility of migrating a locally installed application to a cloud hosted environment with a view to reducing operational costs. In order to evaluate this feasibility, benchmark tests were carried out to gauge the possible benefits of moving legacy applications to the cloud and if increased resources equal increased performance.

Experimentation (in Standalone mode) showed that the effort involved in moving a single instance the Hadoop and Java applications to the cloud were not worthwhile in terms of the benefits shown during the testing phases. Iteration 4 (Pseudo-Distribution mode) displayed similar results, only minimal benefits were documented.

The results from experiments carried out in iteration 5 and 6 showed that once the applications were migrated to the cloud and scaled up (two node Hadoop cluster +), performance increases and benefits began to become apparent. The potentially limitless scale up capability of cloud hosting enabled larger datasets to be tested and as a result the ability to process more data, if required.

The experiments and research carried out and highlighted above have given good insight into the effort required in moving legacy, locally installed software applications to the cloud. The main point that has been highlighted is to do with the importance of carrying out initial benchmarking tests which can then be used as a bar to compare all future deployments with. By following this concept, iterative releases and development phases can be interrogated and judgements can be drawn on them. This would not be possible with a "big bang" style waterfall approach.

3.6 Project suitability for cloud migration

In addition to determining whether or not software is suitable for migration to the cloud, it is also important to understand if a project as a whole is a suitable candidate to be migrated. In order to understand this topic, further research was carried out on this area by Manuja & Manisha, (2014).

What is noted is that agile software methodologies have changed the way that software projects are carried out. The waterfall approach is no longer suitable for this fast paced industry which now requires software developers to react quicker than ever to changing customer and industry trends all of which have been positively impacted on since the rise in popularity of all concepts aligning to the agile manifesto[2].

Since the upward trend in favour of agile methodologies, challenges and risks have become apparent in terms of the systems that support these software development projects. Legacy infrastructures, networks and platforms have come under increased pressure and many gaps and failures have become apparent. These issues are mainly relating to scalability, cost and rapid access to required environments (e.g. staging, test,

[2] http://agilemanifesto.org/

production). As a result of this, companies have chosen to migrate many of their applications to cloud hosted services with a view to overcoming the issues outlined above.

In order to evaluate the pros and cons of migrating an agile software project to the cloud two separate experiments were carried out. “Experiment A” involved a piece of software being developed and tested in an agile environment and then released and retested on an internal server. “Experiment B” involved leveraging Amazon Web Services and deploying and testing software developed in an agile manner.

Once both development and testing cycles had been carried out the results and processes as a whole were examined by technical experts and the results returned. In terms of experiment A, the development and release process was straightforward and no issues were encountered. However, when the software was executed a number of issues became apparent such as; setting up of test environments, resource costs and general server management overheads.

In contrast, by leveraging Amazon`s SaaS, PaaS and IaaS offerings huge benefits were documented such as; straightforward management and availability of testing environments, the ability to increase or decrease resources as required and enhanced continuous integration, testing and delivery capabilities.

Following on from the experiments outlined above a number of conclusions were drawn. In terms of both cost and time the benefits of migrating an agile project to the cloud were quite significant. Both areas were improved greatly by using the services of AWS. For quality, the results were not as positive and it transpired that software and data hosted on the cloud is not as secure as locally stored (on premise) software. This has been highlighted because data such as test plans, test frameworks and test cases may be vulnerable to being compromised as compared to being stored on a local server.

Based on the information reviewed in this article, it can be assumed that there was in fact many benefits (in terms of testing) to migrating existing agile applications/projects

to cloud hosted environments. At the time of writing (2014) it have been fair to say that there were some issues in terms of security in the AWS environment. However, there have been vast improvements in this area and currently security and related vulnerabilities are no longer an issue. With this negative aspect resolved, it may be fair to say that for this scenario, cloud migration was in fact a success and can lead to benefits for agile projects.

3.7 Suggestions and approaches for cloud migrations (from cloud providers)

The majority of mainstream cloud providers have released literature in which they make suggestions as to what they deem to be the correct way to carry out a cloud migration. In 2010 Amazon released a paper which aim was to assist companies in migrating legacy applications from on premise installations to the AWS infrastructure. A number of business and application scenarios were used with a view to demonstrating the amount of effort required and processes to follow for this to be a success (Varia, 2010).

Highlighted by Amazon in the introduction of this paper are three factors that are important when it comes to migrating existing applications to the cloud. They are noted as:

1) "The complexity of the application architecture";
2) "How loosely coupled your application is";
3) "How much effort you are willing to put into migration".

The areas highlighted above are a good guide to follow for any potential cloud migration project because not all applications were developed in a manner which is suited to being hosted off premise and the migration effort required may outweigh the benefits achieved.

In order to assist companies to migrate their applications to the AWS cloud a step by step guide is proposed by Amazon. This guide (see Figure 3.) breaks the migration

process into 6 steps, each focusing on a different area that requires attention during this migration process.

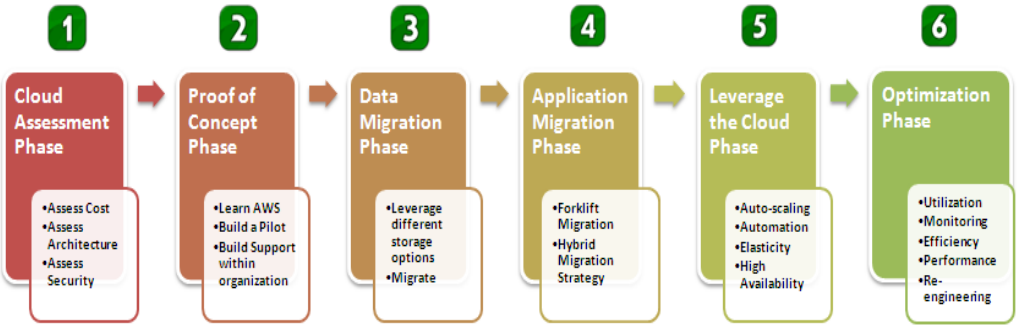


Figure 3 AWS 6 step application migration process, (Varia, 2010)

Amazon then go on to examine and document each of these steps in detail. All steps are well explained and links to all environments/wikis are provided throughout. One of the themes that flows through all papers examined as part of this project is the idea of testing the waters of cloud migration with a proof of concept type approach. Rowe (2013) also suggested this approach as part of a migration framework. As part of this paper, Amazon give detailed steps to carry out this process and it is useful to assist as a starting point for potential migration projects.

As highlighted above, Amazon provide extensive guides to assist companies thinking of migrating existing applications to their AWS offering. Step by step processes are suggested as well as hands on examples. What has not been included in this paper is any in depth detail in terms of software testing and its relevant methodologies. Throughout the document software testing is mentioned but there is no deep dive in terms of the recommended testing approaches, validation of applications to be carried out or software testing cycles that should be ran once the migration is complete.

"Forklift Migration Strategy" is one of the phases suggested by Amazon in this paper and it refers to moving whole applications directly to the cloud in one go rather than iteratively. As details of this approach are explained, Amazon suggests "performing end-to-end testing is a must when using this strategy", Varia (2010). However, no further detail is given on this topic.

The content highlighted above demonstrates that there is a lack of information or interest when it comes to the validation and testing of applications that have been migrated to cloud hosted environments. Considering the detail that has been provided on the migration process, one would have assumed that an extra step should be included which assists with the validation of the migrated application. This paper was released in 2010 and there has been no further iterations to include this specific detail.

3.8 Suggestions and approaches for cloud migrations (from industry and academia)

A collaboration was carried out between the University of Stuttgart and NovaTec Holding GmbH a German IT Consulting company and the aim was to investigate the most suitable and straightforward ways of moving an existing application from an on premise installation to a cloud hosted environment. Once this was investigated, researchers aimed to validate if it would be possible to introduce a vendor and technology independent methodology which could be followed in order to facilitate this change in environment (Strauch *et al.,* 2014).

The application which is being migrated was an enterprise resource management application (NovaERM) based in Java which is used for managing all internal business processes e.g. hiring and administration. NovaTec have recently expanded and as a result the company require a more scalable solution which is not possible with the existing application. As a result, a decision has been made to migrate some, or all of this application to the cloud.

This application consists of a number of individual components each with specific purposes and roles. The application can be broadly separated into three specific layers (Presentation, Business logic, database), each with containing specific standalone components.

For the initial cloud migration, NovaTec wanted to have as little impact on the existing architecture as possible. In a way, this would be a trail to see if this migration was in fact possible at all. To carry this out, NovaTec carried out a study to evaluate the potential options and then outlined the possible challenges that would be faced.

Potential challenges documented:

- Deciding which was the most suitable part of the system to be migrated
- Where to migrate to (i.e. which provider)
- How to determine if the system functions after the migration
- Lack of automation options

The next step in this process was to investigate the potential migration options available in terms of existing methodologies to follow and relevant documentation. Once this information was collected then NovaTec would be in a position to decide if the whole application could be migrated or just the database layer.

NovaTec document a number of potential options that could be followed for this migration process;

- Amazon approach;
- Salesforce;
- A number of different options gathered from various papers.

Based on the information garnered from the above research, it was decided to follow a step by step methodology which consisted of refactoring the existing application with a view to migrating the database layer to the cloud. This migration process was a combination of steps which were gathered from various papers. The steps for migration consisted of seven individual pieces (show in Figure 4 below), each with a set task and goal.

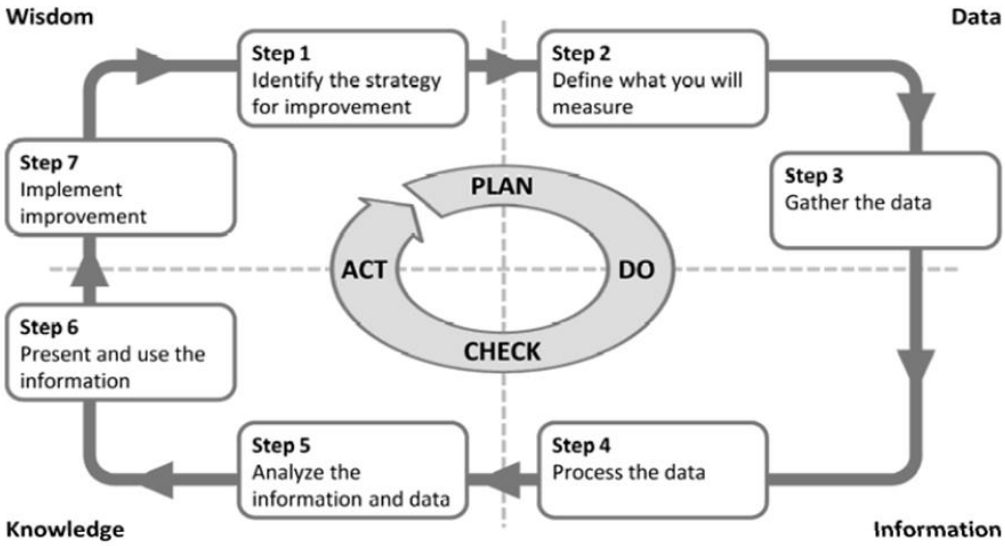


Figure 4. Seven step methodology Strauch, (2014)

In order to test the above methodology, two different types of migration were carried out. The NovaERM application was refactored and the database layer is split out. The database layer was then migrated to a private cloud environment. Secondly, the whole NovaERM application was migrated to Amazon`s public cloud.

Based on this study, it was concluded that the introduction of such a methodology was very beneficial to this process and assisted greatly in migrating to the cloud, regardless of the type of application that was being moved.

What was most interesting about this research was in terms of the amount of time that was spent testing all of the changes that were made to the NovaERM application. At the outset of this research, this was not expected and software testing had not been accounted for in the proposed methodology.

The researchers in this piece of work discovered and outlined “the need to incorporate support for testing and optimisation into our methodology and tool in the future.” Strauch, (2014). This research shows that in a lot of cases when software is being migrated to the cloud, software testing is not even considered. It becomes apparent after the fact when a large amount of time has been spent testing the updated software. This would suggest that in fact, software testing and its cycles have been omitted from the original software migration thought process.

3.9 Migration frameworks

During the research carried out as part of this project, a number of frameworks were proposed for software migrations to the cloud. In a study by Zheng & Du, 2014 a cloud based framework is proposed to assist software developers in migrating client-server systems and applications to SaaS style cloud computing model. The aim was to enable this migration process without the need for costly modifications or the refactoring of code. The proposed framework comprises of four separate modules namely; service proxy, data proxy, tenant management, and cloud resources management. Each of these modules makes up the application as a whole and ensures that all existing functionality remains intact.

Each of the above aforementioned modules aims to guide the migration process with a view to overcoming issues such as tenant configuration, data security and automatic scalability. The actual migration process recommended in this paper is suggested as follows - analyse the original application, create a migration package, and deploy the package to the cloud.

Once the migration is complete, the focus shifts to modularising the application in the new SaaS environment. For the purpose of this case study, an open source CRM application was leveraged. This application was then used to test and validate the suggested framework methodology and associated migration suggestions.

Based on the process outlined above, the migration of an open source CRM application to Amazon`s EC2 cloud computing offering is shown to be relatively straightforward. Through the analysis of this research a number of gaps have been noted in terms of the complexities associated with this kind of development work. The gaps mainly align to the fact that there is little or no verification work (software testing) carried out to prove that this migrated application is performing as expected and that all functionality has remained unchanged.

This type of gap in research has become apparent for many papers reviewed as part of this research project. A number of frameworks and methodologies were reviewed however there is little information in terms of steps to follow in order to verify if the migration has been successful or not. Future work is proposed as part of this paper and aims to focus on performance evaluation and improvements to the framework methodology. This highlights the fact that even in future iterations of this research, software testing such as regression and integration phases are not even factored in to this process it would have been assumed that this would be an extremely important aspect to focus on.

3.10 Conclusion

By carrying out extensive research in the area cloud computing specifically focussing on software testing practices and cloud migration frameworks, it has been noted that there is a lack of information or reference points on this topic. A number of

recommendations have been made in relation to how software should be migrated to the cloud but none that seem to specifically speak of a standardised methodology or framework.

In addition to the above and mentioned previously in this chapter, there are a number of potential options to choose from in terms of cloud migration methodologies. However, in all of the research carried out, there were none uncovered which specifically spoke of how software should be verified and tested once the migration process was complete. There were testing phases highlighted which demonstrated the performance benefits pre and post migration but none that specifically spoke of testing aimed at determining software functionality once the migration had been carried out.

In order to fully understand and investigate the points raised above, the upcoming chapters aim to dig deeper into this topic and clarity to this area. This will be done with a view to gaining insight into how this migration process should be carried out and in what manner.

CHAPTER 4: METHODOLOGY

4.1 Introduction

The aim of this chapter is to describe and document the structured approach planned for carrying out research which aims to provide insight into individual`s experiences of migrating software from on premise environments to cloud hosted implementations. In order for genuine insight to be gained from this research, it is essential that the methodologies and techniques used for this process are robust and well structured.

The purpose of this research is to gather data in a structured manner so that it can be analysed and used to shed light on the research question at hand. By extracting data from individuals who have hands on experiences of migrating software to cloud hosted implementations, it is hoped that new learnings and personal views can be highlighted and documented.

In this chapter, the methods and techniques which will be used for data collection and analysis will be described and set out in order to give clear structure to the chosen research approach. In addition, views and experiences from all contributors will be used to guide the findings of this research project.

4.2 Aim of research

The purpose of this research is to gain insight into the following research questions:

- Which software testing cycles are the most important for software that has been migrated to the cloud?
- What are the differences between software testing on the cloud vs locally installed (on premise) software?
- What issues occur when software has been migrated to the cloud?
- Are software testing tools and methodologies up to date when it comes to software that is hosted on the cloud?
- What existing frameworks and methodologies are available to assist with cloud migration projects?

4.3 Qualitative approach

For the purpose of this of this study, a qualitative approach to data gathering has been chosen. The reason for this is that the aim of this study is to gain insight into individuals' personal experiences (non-statistical) rather than analysing data that is already in existence (quantitative / statistical). This is a largely inductive approach whereby the researcher generates insight and greater meaning to the data that is collected.

By following a qualitative methodology, the researcher is capable of thinking about and studying the experiences of others. This will allow for specific areas of interest to be discussed and as a result, theories and pre conceptions can be verified or disproved. For this particular research project this approach was deemed to be the most suitable because it will allow for free flowing discussions which will assist in more depth being added to the data which is collected.

At the outset of this project, a number of different theories and concepts were proposed. Through the use of qualitative approaches and methods, it will be possible to test and verify these theories and concepts on individuals who have first-hand experience in the areas of interest. This will be a more suitable approach compared to a quantitative methodology which is broader and focusses more on the numerical aspect of data.

4.4 Interviews

For this study, semi-structured interviews have been identified as the optimal method for gathering data which can be used to understand industry experiences and views. This qualitative approach will enable the interviewer to use open-ended questions so that participants can express their views and give insight to their experiences (Creswell, 2003). By following this approach the interviewer can probe more freely into the experiences of the participant and as a result add extra value and depth to the data gathered. This will enable the interview to not only understand the "what" and "how" but to place more focus on the "why" if necessary (Saunders *et al.,* 2009).

In addition to the above, by following a non-standardised approach to interviews, opportunities may arise for discussions to be had in relation to topics that had not previously been considered but which may add significant value to the data collected. This opportunity may arise if the topic being discussed is particularly relevant or interesting to the participant.

With this particular research topic in mind it was decided that structured interviews were not the appropriate option. The reason for this was that these types of interviews are by nature based on predetermined and standardised sets of questions. While a good level of data and insight may be gained by this approach, it would be difficult to determine the participant's specific views if they are not allowed the freedom to describe their experiences with this particular area of interest.

As part of the semi structured interviews there will be a general theme that will be followed and a number of key questions that will hopefully be covered. By making the interview process non-standardised it is envisioned that each interview may vary depending on the participant. This will allow for a certain level of dynamic interaction depending on the experience and input of each participant, Saunders (2009).

The format that this process will take is likely to consist of a brief description being given at the outset of the interview in relation to the purpose and goal of the interview. By doing so the participant will be aware of the overall aim and as a result ambiguity will be at a minimum. All conversations will be recorded so that all data will be captured and analysed.

There are a number of risks aligned with following a non-standardised approach to interviews. Issues may occur if the participant is not comfortable with being recorded or if the topics being discussed are private or secret to his or her organisation. In order to mitigate against this problem, research topics, themes and interview questions will be shared prior to the interview. This will allow for the participant to prepare in advance of the interview so that the questions or topics that may arise do not come as a surprise to him or her.

The aim is to conduct all interviews in person, if interviews are carried out face to face then there is a better opportunity for a rapport to be established between the participant and the interviewer. By conducting face to face interviews there is a greater chance of information being exchanged freely and more of a conversational flow will be established. In contrast, if this information were to be gathered via questionnaires then there would be a reluctance for the participant to provide explanatory or in depth answers. This may also lead to data being diluted and possibly irrelevant. It may not be possible to conduct all interviews in person, if this is the case then a phone call or Skype meeting will be carried out, Saunders (2009).

The length of each interview will vary depending on the conversational flow, relevance of information being provided and schedule of the participant. Overall, this will be a time consuming process which may lead to varying quality levels of data being collected depending on the uniqueness of each interview.

4.5 Sampling and selection

In light of the specific research question at hand and bespoke discussion points, it was decided to follow a purposive sampling approach for choosing potential participants for this study. The reason for this is that a judgement needs to be made in advance of the interview as to whether or not the participant will have the knowledge required to answer the main research question and meet the overall objectives.

Due to the time constraints of this project and the time intensive nature of interviews in general, purposive sampling was chosen as a suitable option when working with small groups of samples. The aim was to focus on individuals who it was hoped would have particular informative information based on their specific experiences and technical competencies, Saunders (2009).

In order to ensure that suitable participants were chosen for this study a number of inclusion criteria were established. The purpose of this approach was to maintain a standard of experience and competencies in the computing field and as a result the quality and relevance the data collected could be maximised. The inclusion criteria consisted of the following headings;

4.5.1 Job Title

This was an important criteria that needed to be set because it was essential that all participants involved in the interviews had experience in technical roles that were in line with the research topic overall. The main aim was to select participants that were heavily involved in software development, QA or solutions designers (specifically cloud solutions).

By focusing on these types of IT professionals it was envisioned that greater insight would be gained into the data collected. In addition, it was important to make sure that the participants had hands on experience with cloud based solutions.

4.5.2 Years of technical experience

The reason for including this criteria was that in order for valuable data to be collected, the participant must have a substantial level of technical experience and knowledge. This is important because without industry and hands on experience the data collected may be unsuitable or lacking in technical insight. Five years has been set as the minimum technical experience required to be included in this study.

4.5.3 Qualifications

In order to gauge technical competencies, it is a requirement of this study that the participants involved should have some technical qualifications. This criteria is included so that a rounded interview can be carried out and that all participants are at a similar level in terms of knowledge or technical experience. If in fact a participant does not have an academic background then 10 years of industry experience is required to participate in this study.

Participant	Job Title	Years' experience	Qualifications
A	Polaris Developer	15+	
B	Scientific Consultant	5+	Master of Science, Computing

C	QA Engineer	5+	BSc. in Business information & technology systems
D	Senior Test Analyst	5+	Bachelor of Science, Business Computing
E	Consultant Engineer	10+	Higher Diploma, Computing

Figure 5. Participant details

All participants in this study were sourced through personal and professional contacts of the researcher. Priority was given to participants who had personal experiences of working with cloud based technologies in a software development or QA environment. In order to get a sense of the calibre of each of the participants, a profile of each is presented in the upcoming section.

4.6 Participants

Participant A is from Scotland but currently living in Co Roscommon. He has over fifteen years' experience of building industry standard insurance software. This software is used for rating insurance products of all types such as; motor, home and commercial. Throughout his career, participant A has participated and led a number of high profile software projects with one of Ireland's largest insurers. He is highly proficient in software coding and testing, specialising in VB, Productwriter and .Net as well as hands on experiences with cloud computing.

Participant B originates from India but has lived in Ireland for over five years. He has an MSc in Computer Science and over five year's computer programming experience. He has worked as a software developer for one of the world's largest networking companies as well as building software for the payments industry during other roles.

Throughout his career, participant B has worked with cloud computing technologies in multiple forms and on multiple projects.

Participant C is from Co. Mayo and is a QA specialist for a large reinsurance company in Ireland. He has seven years' experience extensively testing software in a number of roles in industries such as financial, insurance and medical devices. In his last two roles, participant C has been heavily involved in the testing of software that has migrated from local servers to cloud hosted environments. During these projects, this participant has gained deep insight into the benefits and challenges of cloud computing.

Participant D is from Co Westmeath and has spent his whole adult career working as a QA (6 years). Throughout this time, he has moved from a junior QA role to a team lead on a large scale project within one of Ireland's largest insurance companies. As part of his current role, participant D has played a key role on a cloud migration project, this has allowed him to gain insight into this process from a high level and deep dive perspective.

Participant E is from Co. Dundalk and is currently working as a software consultant in Dublin city centre. His previous role consisted of electronics and software development with a large hardware multinational. Participant E is currently studying for a higher diploma in Science in Computing in DIT.

4.7 Organisation profile

Participant A works for a large insurance company with 1,200+ employees based in Ireland. This company is over 100 years old and until recently its business focused on insurance and not technology. In recent years this focus has shifted and emerging technologies have been embraced and began to be embedded into insurance products.

Participant B currently works as a scientific consultant for a large accountancy firm in Ireland with over 1,000+ employees. Prior to this role, participant B was a software developer for one of the world's largest networking companies in the world. This

company specialise in implementing large scale networking systems and manufacturing all of the associated hardware and networking components.

Participant C currently works as a QA engineer for a multinational reinsurance company. This company are located in Ireland and worldwide and provide trading and insurance software. Participant C`s previous role was with a healthcare technology provider who build software solutions for the healthcare industry with 100+ employees located in Ireland.

Participant D currently works for a large insurance company who employ 1,000+ people in Ireland. This company are currently focussing their efforts on migrating away from legacy systems to newer technologies. In a previous role, Participant D worked as a test engineer for a large software house based in Dublin. This company built software solutions for the financial industry.

Participant E also works for a large accountancy firm in Ireland with over 1,000+ employees. His role is a consultant engineer with large multinational software companies. Prior to this role, Participant E worked for multinational microprocessor manufacturer with over 4,000 employees in Ireland.

4.8 Data collection

The interviews and data collection for this study took place in May and June 2018. As mentioned above, all interview questions were sent to participants in advance of each interview. A laptop and iPhone 7 were used to record all face to face conversations and the results were then transcribed. Each of the participants were initially contacted via email and then a phone call in advance of the interview. All interviews were carried out in cafes close to the participants' workplaces.

One of the benefits of carrying out semi structured interviews is the freedom that the interviewer is allowed in terms of the questions and structure followed, this may vary depending on the responses and general feeling or rapport at each interview. The questions that will be posed will mainly be open ended, this will allow the participant

to give their specific viewpoint and experiences. In addition, if topics of specific relevance to this research are brought up, then more probing questions can be posed.

In order to ensure efficiencies, each interview will begin by the researcher explaining the format and purpose of the interview to the participant. This will be then be followed by a discussion based on the interview questions posed which are broken up into the following sections.

The interview schedule was structured into the following five sections:

1. Technical/educational experience.
2. Experience with cloud computing.
3. Experience migrating software to the cloud.
4. Issues encountered.
5. Recommendations.

The aim each section and the interview questions in general was to guide the conversation as to the relevant topics rather than a structure that must be followed. In advance of this data gathering exercise, a mock interview was carried out to ensure that the questions that were to be posed were straightforward and that the interview flowed as expected. This allowed for any teething problems to be ironed out so that each interview could be as efficient as possible. This data was not included in the final submission of this project.

4.9 Data analysis

Once all interviews are completed the data will be transcribed, coded, analysed, verified and interpreted. In order to classify and categorise all data, the recordings will be listened to on a number of occasions with a view to identifying recurring trends, key words or shared practices. If any themes or common information become apparent, the data will be categorised accordingly and the researcher will be capable of identifying and verifying trends from all data that is correlated, Saunders (2009).

This data will then be used to compare findings highlighted in the literature review chapter above as well as identifying if there are any unexpected trends the warrant further investigation. By following this process, insight will be gained into the shared

experiences of all participants. These findings will then guide the upcoming chapters by highlighting any expected, unexpected or recurring themes.

The final stage will involve drawing conclusions based on the information provided by all participants and the information available in literature from both academia and industry.

4.10 Ethical considerations

Prior to this study commencing, a research proposal was submitted and ethical approval was provided by the course coordinator. Considering that this is a research based project, it is paramount that the research gathered is of high quality and integrity and gathered in a manner that is ethical and transparent (Paradice *et al.*, 2018). It is essential that this consideration flows throughout this research project.

The aim of this research project was to gather relevant information in the area of cloud computing. This information will be collected via participants and information which is currently available. The information will then be collated and analysed with a view to gaining insight and understanding.

All participants in this study have been identified based on their technical and educational experience. Prior to this study commencing, each of participants were contacted individually and a list of interview questions were circulated. By following this approach, the participants had an opportunity to process all of the questions and decide if they could add insight into this area. They all participated on a voluntary basis and were free to withdraw at any stage.

Each of the interviews carried out as part of this study were recorded and the participants were informed of this accordingly. At the outset of each interview the purpose of the meeting was explained and the participant had the opportunity to ask questions as required. The interview questions posed in this study aimed to suit participants of varying technical experience and skill while being impartial and independent. The aim of this was to give each participant the freedom to freely discuss their own specific experiences. In addition, there were under no obligation to divulge

any sensitive information and were not required to answer all questions if they did not feel they were appropriate.

4.11 Limitations of the study:

During the process of carrying out each interview a number of limitations emerged, this has mainly become apparent because of the small number of participants interviewed. In order identify, contact and interview five participants, twelve individuals needed to be contacted originally. Following discussions with each and mainly due to time constraints (with the potential participants), not all were suitable or available.

In addition, the participants who did participate in this study were also under time pressure and did not have unlimited time to give to this interview process. As a result, the aim was to only pose relevant questions so that the time with each could be optimised.

While conducting any type of research, the more information that is captured the better and this project was no different. It transpired that although the semi structured interviews, transcriptions and coding process was time consuming in-depth information was indeed collected. It is also important to note that bias from both the researcher and the participant aimed to be avoided at all costs from this study. This aimed to be achieved by not including irrelevant questions, not making assumptions and ensuring clarity and integrity from the questions posed and answers given.

The information gathered for this study was done using qualitative approaches and this information was then interpreted and conclusions were drawn. This approach was deemed to be the most suitable for the research at hand, a numbers based approach (i.e. quantitative) nay not have given the same level of information and insight.

4.12 Conclusion:

The aim of this chapter was to describe the systematic and planned approach taken to gathering data for this research project. The aim was to use a methodology which

could provide the greatest insight into the research questions posed. By following this approach and ensuring that high quality data is collected it was envisioned that a deep level of insight into the topic at hand can be gained.

CHAPTER 5: FINDINGS

5.1 Introduction

The aim of this chapter is to analyse, scrutinise and identify the main themes identified throughout the data that was collected as part of this project. In order to so the transcribed and coded interview data was analysed with a view to determining if there was in fact any emerging, expected or unexpected themes becoming apparent. By doing so, it may be possible to create correlations, commonality and recurring patterns between all participants.

The purpose of carrying out multiple interviews was to gather data that was both relevant and thought provoking. By doing so it was envisioned that the data collected could be used to gain insight into a number of IT professional's hands on experience with migrating software to the cloud and the associated testing practices carried out. This will allow for a more practical (rather than theoretical) representation of their cloud migration experiences.

5.2 Interview organisation and feedback

As mentioned previously, prior to each interview all participants were sent the final list of questions which would be discussed in depth during each of the interviews. An interesting point to note is that when each of the participants were contacted initially regarding the possibility of participating in this research project, they each specifically noted that they may not be able to add much insight into the area of cloud computing.

However, once the questions had been sent and the participants had time to think about the specifics of this research project they each in fact had excellent hands experience and valuable insight in the area of cloud computing. This was interesting because, prior to this project each of these participants were identified based on their vast technical experience in the areas of QA and software development. This was in fact the case and the interviews progress well once the participants had time to reflect on their own experiences with cloud technologies.

5.3 Previous migration experience and organisation culture

Based on information collected from each of the interviews carried out as part of this project, a general sense and cultural viewpoint in terms of practices and methodologies followed of each of the participants' organisations was captured. Following each interview, it transpired that each participant and in turn their organisation had varying experience in terms of their exposure to migrating software to cloud hosted environments in general.

Some had varying experience of carrying out software migrations internally between local environments and servers however, the hands on experience of migrating to the cloud varied drastically. Each participant and their past and current organisations were on different parts of the road map in terms of their transition to the cloud.

As mentioned previously, participant A works for a large insurance company who have in recent years shifted their focus to embedding the latest IT technologies into their existing insurance products. As a result of a slow start in terms of technology uptake, this company are playing catch up and investing heavily in technology.

They had previously carried out projects to attempt to simplify many of their existing insurance systems; this type of work was focused on local applications which were hosted on internally managed servers. As described by participant A
"*we have our rating engine which we use in the insurance system is a packaged software that we deploy on to a host server*".

In line with this appetite for embracing technology, cloud computing has been identified as a potential solution to some of the current issues within this company. Participant A stated that
"*It (i.e. the cloud) is something that will definitely be changing our roles in the future*".

In order to begin the transition to cloud hosting, this organisation are beginning to migrate some of their existing applications in an iterative manner, as described by participant A

"*The process was worked through in a step by step basis to enable the testing in small increments*". Participant A was involved in this process and this formed the basis of the interview with him.

Similar to Participant A, participant D works for another large insurance company in Ireland. This company are in a similar position to the previous company mentioned as in they are now investing heavily in IT technologies after many years of focussing only on their insurance business. As described by participant D
"*ABC would be a reasonably old company so they would have used a lot of legacy systems and nearly archaic ways of working. So they are now starting to move into modern technologies*".

Participant D`s company took an alternative approach to migrating legacy technologies to the cloud. Their approach was to carry out a "big bang" move from a local environment to an AWS managed cloud, which in hindsight as described by participant D was not the most productive or straightforward approach to take,
"*so I would recommend doing an incremental migration, so (migrate) individual components rather than trying to deploy all of the components at once*".

The experiences and industry culture of participant B`s organisation were vastly different to that of the two participants highlighted above. As mentioned previously, the organisation that this participant works for is a long established worldwide IT networking company who have carried out numerous software migrations to the cloud and are vastly experienced in this area.

As a result of this, participant B`s organisation have extensive experience in migrating legacy applications from on premise locations to cloud hosted environments. This vast experience will form the basis of the interview with participant B and insight will gained into this area. An example of this experience as described by participant B is "*we have project that were taking decades old (20 years +) applications and moving them to the cloud*".

Participant C has been involved with the testing of software that has migrated from an on premise location to the cloud in his last two roles. His past role was as a QA analyst with a leading provider of healthcare software for Hospitals, Consultants and Pharmacists. There was a culture of innovation at this company and new technologies were embraced and investigated with a view to understanding if they were a fit for this organisation.

In line with this culture, cloud hosting was investigated as a potential option for migrating locally installed software to the cloud. This migration is described by participant C as follows
"*We went from being a locally installed product that involved a lot of effort to install*" to software being "*hosted on the cloud so that it is not installed on anyone's machine*".

In addition to the above, participant C`s current role is as a senior QA analyst with a reinsurance company in Ireland. As mentioned previously, this company are a leading global provider of speciality reinsurance software products. Due to the fact that this is a worldwide company, they have embraced cloud technologies and heavily leverage the capability of the cloud to provide accessibility to its software regardless of a users' geographical location,
"*we would use a lot of cloud computing in my current role*". As a result of the geographic location of all users of this software, cloud computing is at the backbone of this company`s philosophy.

In light of the above, it has been noted that all participants in this study have varying levels of experience in dealing with cloud technologies in general. The purpose of interviewing these specific participants was that they could each give their experiences from different industries and from organisations who are at different stages of cloud adoption.

5.4 Themes

While analysing the data collected during the interviews for this research project, a number of separate themes became apparent. These themes were noted as a result of commonality in the answers that were provided by each of the interview participants.

Although each specific theme was not noted or documented prior to the interview process, they were not unexpected and were in line with the research topic at hand and the questions posed. These themes are highlighted, described and investigated below.

5.4.1 Reasons for migrating to the cloud hosted environments

What became apparent following each of the interviews carried out was the reason why participants and their companies had migrated from on premise deployments of their software to cloud hosted environments. This migration was required because of limitations in the existing software and its host environment.

Participant B noted that there were significant challenges in terms of performance with existing software applications and the fact that it could not be accessed by a worldwide user base while installed on the current environment.
"*We came across a challenge whereby we have blockages or bottlenecks in software when it comes to handling multiple queries*", "*we want to have an option for other teams working globally to have access to the cloud and the resources*".

This suggests that multi tenancy would solve this issue and that when these applications are moved to the cloud they will become accessible to all users, regardless of their geographic location. This functionality may not be accessible with an on premise server.

In addition to the above, participant D`s organisation embarked on a project to migrate an existing CRM tool to the cloud so that it could be used and accessed by its
"*agents and to the branches across Ireland*". This "*CRM tool that we had hosted internally" was "migrating that up on to the cloud to store and to host*". This suggests that the environment on which this tool was previously hosted did not offer this functionality.

5.4.2 Requirement for more in depth testing cycles

Based on the discussions with each of the participants in this project it became apparent that there are some differences between the testing cycles and practices that

are carried out on software that is locally installed (on premise) and software that has migrated from a local environment to the cloud.

Initially, this theme emerged in the interview with participant C who stated,
"*You have to really, really aggressively test it (i.e. software that has migrated to the cloud). A lot of negative, edge case testing. Where a system would perform in one way if it is locally installed whereas if it was cloud based it may not have the support there*". This statement suggests that there are a lot of extra scenarios that need to be considered when existing software has been migrated to the cloud.

In addition to the above, participant C also mentioned that by migrating software to the cloud the capability to create multiple instances of software is now possible. This has begun to happen following requests from customers to create bespoke applications to meet their ever changing needs,
"*While in the new system (i.e. the cloud), it was very easy to just create a new branch specific to that user and give them their own tailored version of the program*".

This capability was not possible with the previous application,
"*If 10 different customers want the system to do 10 different things, it was very hard for us to program that into our old system*". This suggests that by moving software to the cloud and by enabling increased functionality and capacity as opposed to moving it internally to a bigger or higher capacity server the testing scenarios and test cases to be ran on the software has increased exponentially.

This theme also became apparent in the interview that was carried out with participant D. He stated,
"*We had to do some increased performance testing and we did more in-depth testing cycles after we had migrated it (i.e. the software) just to verify that it (i.e. the cloud) would be fit for purpose and that it was all working correctly*". This suggests that this increased more in-depth testing had become a requirement for this software after it was migrated to the cloud. This increased level of testing was not necessary when the software was installed locally.

Following the migration of a locally installed CRM tool to the cloud, participant D noted that

"*a number of components failed to work as they had previously (i.e. prior to the migration").* This would also suggest that further testing cycles were required to determine if these components were functioning as expected once the migration had been completed.

5.4.3 An iterative approach to cloud migration

A theme that flowed between all interviews carried out as part of this project was that of iterative or staggered migrations from local environments to cloud hosted implementations. Each of the participants mentioned this as an important factor in the transition between both environments. It was noted that this had not been mentioned in relation to internal migrations (e.g. server to server), it mainly related to cloud migrations. As described by participant E

"if a business critical requirement needed our space then we were pushed it (i.e. the software) around internally then as needed".

Participant A noted that,

"The exact same testing where at all possible was carried out, it was the same software that we were deploying before and after the cloud and the in house server". He then however stated, "The process was worked through in a step by step basis to enable the testing in small increments, as and how things were being set up and deployed".

This suggests that the original testing phases which had been used for local installations had to be updated to support an iterative software migration to the cloud because prior to this, this was,

"packaged software that we deploy on to a host server". In contrast, if these testing phases are only being updated for this cloud migration process then they were not required for previous internal migrations that were carried out within this organisation.

Participant C noted that

"You have to literally get the basics (of the software) up there, like the straw house. Literally, like the least amount of software that you can use to test and you have to expand that a bit more". What is being suggested here is an MVP[3] type approach whereby the simplest iteration of the software is migrated to the cloud with the aim of determining if the basic functionality works as expected once migrated.

This type of an approach is different to a straightforward migration between internal servers within an organisation whereby the software would need to be architected in a manner so that the entire system could run on different types of application servers and function regardless (Wang & Yang, 2014).

If this type of migration between servers was being carried out, the environment to which the software was migrating would need to be similar in terms of its capability and configuration; the main change may be its capacity, performance or location. Whereas with a migration to a cloud hosted environment, the set up and connections may vary drastically and the configuration is managed elsewhere.

The reason for this type of approach is that the viability and compatibility between the software that is being migrated and the cloud environment needs to be established prior to the whole software application being migrated. Through his recommendations, participant D provided good insight into this approach

"Yes, so I would recommend doing an incremental migration (to the cloud). So individual components rather than trying to deploy all of the components at once. I feel if we had done more incremental components it would have resulted in our testing being a bit easier and more focussed and streamlined".

The above points are important to note because based on the information gathered in each interview, the participants gave the impression that this was a new way of doing software migrations. It became apparent that migrating software to the cloud was in fact not a straightforward process and a large amount of new learnings were required, as described by participant E

[3] https://www.agilealliance.org/glossary/mvp/

"I think you would really need to plan it out (i.e. the migration) a bit more and perhaps do it in stages".

5.4.4 Types of software issues encountered during/after the migration process – "Wiring".

The aim of this research project was to identify and understand the types of issues that become apparent through the testing of software that has migrated from an on premise environment to a cloud hosted environment. One of the themes that became prevalent during the data analysis phase was that there was in fact many issues encountered by participants who had been involved in this migration process.

Participant C classified the issues he encountered following the migration process as "wiring" issues. This in fact was a good phrase to demonstrate a commonality between responses from all participants. For participant C this issue manifested itself in a testing phase which identified that
"some of the ports were incorrectly networked together" on the new cloud hosted environment. If this had went unnoticed, it would have resulted in significant issues for customers. This issue may highlight the fact that when you are outsourcing or moving to an externally managed environment there may be issues that are encountered which would not happen if the environment was managed by internal employees.

Participant A described issues encountered in the initial integration phase of testing which highlighted that in fact a number of calls between various systems and front ends were not functioning as expected and communications with the new cloud URL were incorrect. This was in addition to
"teething issues here and there with various different response times and calls to different databases" post the migration phase.

What became apparent as part of this common theme was the fact that these types of issues seem to be in relation to the environment in which the software has been migrated to i.e. the cloud. They were discovered because the new cloud environment is now accessible by more users in new locations worldwide. This capability was not

previously achievable with the existing on premise implementations. As described by participant B

"whereas now, we have accessibility issues because the system is being accessed throughout the world".

5.4.5 Types of software issues encountered during/after the migration process – Limitations of functionality of existing software.

Another theme in the context of issues encountered during and after the migration process that became apparent was in relation to the fact that software which was migrated to the cloud did not function as expected once the migration was complete. Participant B noted that once the legacy software was migrated to the cloud and placed under increased load (in terms of user traffic) that the software was negatively affected.

"I think it was to do with the new environment because the old system doesn't really support these kind of rigorous tests and operations".

Participant C stated that there were issues logged as a result of the move to a new cloud environment from a local server. In his case,

"There was issues at the start because with the ramp up/transition to cloud based software there was an almost immediate rise in the number of customers, which doubled or trebled overnight". This had not been an issue prior to the migration because the software was not exposed to such user traffic spikes.

In light of the above, there was a sense from both participants that it is important to fully understand the existing capability of software before it is migrated to the cloud. This became apparent for each of them because their existing software proved to be incompatible with the increased performance demands that were put on it once migrated.

5.4.6 No specific framework or methodology followed for migrations.

Due to the fact that a number of the participants and organisations who contributed to this study were relatively new to migrating software to the cloud it would have been assumed that they would have taken guidance from their cloud provider on how best to carry out this process. All participants involved used either Amazon or Microsoft, both of which supply in depth step by step guides as to how best to migrate legacy software to the cloud.

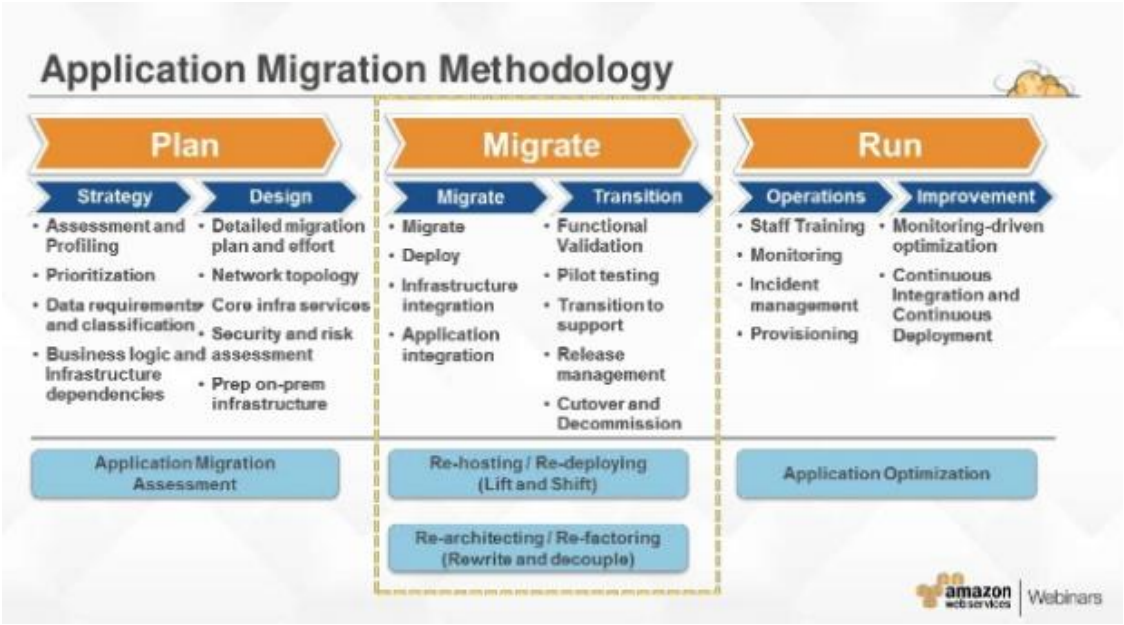


Figure 6. Migration Methodologies (Amazon Web Services, 18:58:19 UTC)

However, this transpired not to be the case and in fact there were no commonality in the frameworks or methodology's followed. Participant B`s organisation followed an Agile or Lean approach whereby they

"work on a feedback basis and try to do one step at a time and then test it and then work on the enhancement aspect at the same time as well". This is more of an iterative development approach rather than an approach for migrating to the cloud.

In participant A`s case, his organisation followed a similar iterative approach whereby they moved the legacy software over a

"step by step basis to enable the testing in small increments, as and how things were being set up and deployed". This migration seemed to have been carried out iteratively

as well and with some level of caution in order to ensure that the software functioned as expected once migrated.

These points are interesting to note as it would seem that there is no high level plan as to how this migration will happen in terms of a strategy – design - migrate etc. similar to what is described in Figure 6. At the outset of this project, it would have been assumed that there would have been an overarching roadmap as to how large scale software migrations like these would have taken place within organisations. This is an alternate approach to that of moving software internally between servers, which may happen in a very planned and structured manner and can lead to work intensive processes (Link *et al*., 2006).

5.5 Emerging themes

Prior to this projects initiation, there were a number of topics and themes which it was envisioned would be discussed and investigated in line with the commonality in terms of experience of all interview participants. However, after carrying out extensive analysis on all interview data, a number of unexpected themes became apparent, details of which will be investigated below.

5.5.1 No real knowledge of SaaS, PaaS, IaaS or external dependencies

One of questions posed to all participants was - Have you developed or tested software on the cloud? And if so, in what form - SaaS, PaaS, IaaS? At the outset of this project it was assumed that any technical users (such as the participants) of cloud hosting services would be fully aware of the differences and terminology in terms of the implementation of each of these cloud variants.

However, it transpired that in fact this was not the case. When the question was posed to participant A he stated, after a significant pause
"It's a software as a service implementation as far as I am fully aware." The impression given was that in fact he was not sure of the specific type of implementation and just answered the question using one of the suggestions that the interviewer had provided.

In addition to the above, when the same question was posed to participant C there was also a long pause and he answered

"Yes, it was mostly software as a service" but in hindsight, a certain level of confusion came across and there was no deeper insight into the specifics of the implementation.

The above theme highlights a gap in knowledge in terms of the types of cloud implementations that users are deploying there software to. This is interesting because it suggests that users are not fully aware of the external dependencies to which their newly migrated software is at risk or exposed to. This may mostly become apparent in a SaaS or PaaS implementation whereby the cloud provider is fully in control of the vast majority of the functionality or configuration of the cloud.

A scenario may arise whereby the provider updates or changes some of the underlying configurations. This may result in unexpected clashes or mismatches with the software that is installed on this environment. A system outage or unwanted changes to existing software functionality may be the result of this update. If the cloud providers changes are unexpected and ongoing then constant patching and updating may be required which is time intensive and costly.

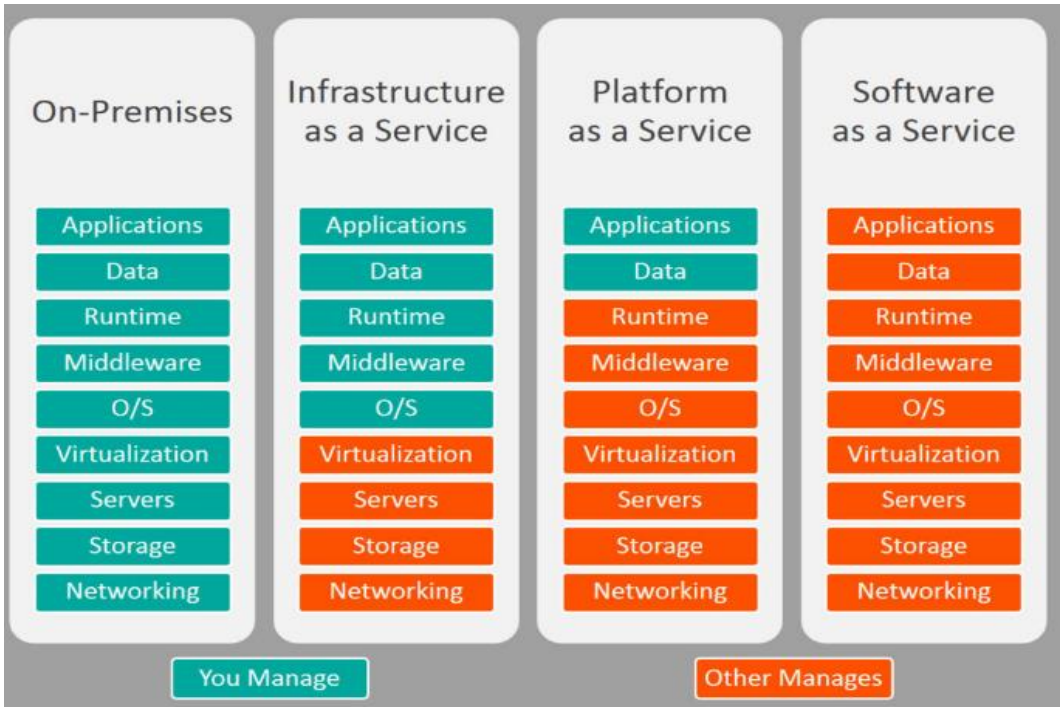


Figure 7. Cloud implementation variables ("SaaS vs PaaS vs IaaS: What's The Difference and How to Choose – BMC Blogs," 2017)

5.5.2 No standard software testing tools used

Due to the fact that large scale cloud service providers are limited to a handful it was envisioned that the majority of participants in this study would have chosen either Microsoft Azure or AWS as their cloud provider. Following analysis of the data collected this was in fact the case.

Once this was clarified, it was then envisioned that there would be some level of commonality in terms of the software testing tools used to verify that software migrations to the cloud were functioning as expected. Each of the large scale cloud providers have recommendations for the types of tools and technologies that should be used to verify the functionality of legacy software that has moved to their environment, as demonstrated in Figure 8.

Figure 8. Azure migration partner tools ("Azure Migration Centre – Cloud Migration Planning | Microsoft Azure," 2016)

Based on the data collected from each of the interviews carried out as part of this project, this was not in fact the case. The majority of participants interviewed used bespoke in house tools to verify and test their software once it had been migrated. Participant B stated

"We prefer to use in house tools for testing and making sure that functionality is ok".

This was mainly due to the fact that *"we don't really use external tools for security reasons"*.

Participant A was in a similar position whereby his organisation use “GUI tools that we have designed ourselves”. These tools work in conjunction with performance and automation tools such as JMeter, QTP and UFT. For this organisations specific scenario, these tools were deemed to be the most suitable. As highlighted by participant E,

“testing for this particular project involved running manual scripts so and manual testing to check the functionality”.

The consensus in general throughout this project is that there are no go to software testing tools for verifying the functionality of software that has been migrated to the cloud. Each of the participants described why their specific needs warranted a specific tool and as a result, no commonality can be drawn upon.

5.6 Conclusion

The purpose of this chapter was to examine and investigate the findings of the data that was collected as part of this project. Based on the above information, it has become clear that there are in fact some distinct differences between the software testing practices and techniques that are carried out on local installations of software and that of software that has migrated from local environments i.e. on premise local to cloud hosted environments.

In addition to the above, what has also been highlighted is that fact that based on the information that was collected and analysed from each of the interviews carried out, migration practices and methodologies or frameworks followed are non-standard by nature, regardless of the recommendations made by large scale cloud providers. This is an interesting point to note as by and large, the environments to which organisations are moving their software is largely similar in terms of configuration and functionality.

CHAPTER 6 DISCUSSION

6.1 Introduction

The main aim of this research project was to understand and examine the testing practices that are carried out and required for software that has migrated from an on premise environment to a cloud hosted implementation and if this migration process has followed a specific methodology or framework.

This chapter will provide insight into the results obtained and determine why the data collected is relevant to the research topic at hand and how it correlates with the previous research carried out as part of this project. The results and information collected as part of this study are based on data from 5 semi structured interviews with IT professionals who have had hands on experience with migrating software to the cloud.

6.2 Understanding the results of the data collected

As expected, each of the participants interviewed as part of this project had varying degrees of experience with cloud computing and the technologies aligned with it, each had carried out migrations of software to the cloud. The difference between them was the fact that the organisations in which they worked had adopted cloud hosting at different times and as a result, the maturity of the cloud migration projects varied drastically.

6.2.1 An Agile approach to software migrations

What became apparent following the analysis of data collected as part of this study was that in fact there was no specific software migration methodology or framework followed by any of the participants. This was interesting because although all participants had experience with cloud technologies they were relatively new to migrating commercial software applications from local environments to the cloud. With this in mind, it would been logical for these participants to follow

recommendations from the likes of Amazon or Microsoft as to how to best carry out migrations.

As discussed earlier, this was not the case. One particular reason for this may be the fact that with modern software companies and developers in general there is an element of giving something a go and seeing if it works as opposed to following a structured roadmap, Manuja (2014).

This is the kind of mentality that has been introduced by agile methodologies and enabled by the cloud. Developers can now come up with ideas, develop a POC and launch it to the cloud. As shown in this project, this process has also become apparent in software migrations to the cloud. Organisations are in a way following their own specific migration frameworks which is an alternate approach to that of the sequential nature of the waterfall methodology, Manuja (2014).

6.2.2 Limited investigation into legacy software prior to migration

As noted previously in this report, a number of participants remarked on the limitations of their existing software once it had been migrated to the cloud. This is interesting to note because, if software had been identified as a potential option for a cloud migration project then its functionality may be suitable to handle increased loads and pressures. Obviously the reason for moving to the cloud may be to open the software up to an increased user base but this functionality could have been incorporated prior to the migration.

It may be difficult for developers to estimate the volumes of loads the newly migrated software will be put under and if in fact in can cope without decreasing functionality Rowe (2013). By fully understanding the capability of existing software its suitability for migration to the cloud may be judged more clearly and as a result, be not left lacking once the migration is complete.

6.2.3 Increased testing phases required once software has been migrated to the cloud

What has become apparent following the research highlighted above is that once software is migrated to the cloud increased testing and verification work is required by all involved. The reason for this is the fact that in a number of cases, various issues became apparent during the migration process which highlighted the need for increased testing on the newly migrated environment. In addition, the move to a new hosting environment may have unknown impacts on software.

Prior to the migration to cloud hosted environments, software which was locally hosted may have had a number of testing phases built up around it and structured processes around how changes are verified. The move to the cloud may now have changed this structure and as a result a level of uncertainty is apparent as well as the need for increased research into how the software will be tested in the new environment (Murthy & Suma, 2017).

6.2.4 Understanding the dependencies and relationships with cloud providers

Throughout this research project, a lack of clarify was noted from all participants in relation to the types of cloud variants available and to which they are migrating their software to i.e. SaaS, PaaS, IaaS. This in conjunction with the fact that no cloud provider migration methodologies were followed suggests that users of cloud services are somewhat unaware of the environment specifics to which they are migrating their software to.

This may be the case because with cloud computing in general the cloud providers have full responsibility of the underlying systems and configurations. As a result of this, users may assume that all systems just work regardless and any required changes or updates that the provider needs to make happen in the background.

Due to the fact that the majority of participants and organisations noted in this study are relatively new to cloud adoption, the above assumptions that everything just works may be apparent. As the maturity and complexity of this software grows, so will its

ties to the cloud. In the future a scenario may arise when there is a conflict between a change made by the provider and the hosted software (Taha *et al.*, 2017) and as a result, SLAs may need to be examined more closely.

6.3 Conclusion

As described above, a number of discussion points became apparent based on the data collected in each of the interviews carried out as part of this study. It is interesting to note that while each of the participants involved is at a different stage in their migration to the cloud, they all have some shared commonality in terms of the issues and challenges faced.

CHAPTER 7 RECOMMENDATIONS

7.1 Introduction

In this chapter and based on the findings and discussions above, a number of recommendations will be made in relation to the testing of software that will be or has been migrated to a cloud hosted environment. The recommendations put forward in this chapter are based on data collected from relevant interviews and research carried out as part of this project.

7.2 Validation of legacy software prior to migration

As highlighted above, it has become apparent following interviews with a number of participants that once they migrated legacy software to the cloud issues occurred in terms of the performance and functionality of the software. These types of issues mainly became apparent once increased user and performance loads were place on the software.

In order to mitigate against this type of issue, it may be worthwhile for users to carry out pre migration checks on software that they are planning on moving to the cloud. This could be achieved by following a sequence of logical steps to evaluate whether the software in mind is a suitable candidate for migration to the cloud and if so, the potential benefits (in terms of performance) that it might achieve, Rowe (2013).

This could be done by tracking the amount of users that currently access the software and then estimating the user base once migration to the cloud is complete. By carrying out these checks and pre-empting the move to the cloud, functionality or increased capability could be built into the software which may lead to the avoidance of future performance issues on the cloud.

7.3 Methodologies and frameworks

As discussed previously, during this research project it became apparent that none of the participants involved used a standardised migration methodology or framework. This was largely due to the fact that migrating software to the cloud has many different variants depending on the organisation involved, the specifics of the software and the experience of all involved.

For this reason, it is difficult to implement or suggest an umbrella or catch all migration strategy for each specific use case. However, it may be useful for organisations to formulate a project specific migration strategy prior to initiating a software migration project. By taking the time to formulate a suitable strategy which takes into account the recommendations for the chosen cloud provider or expert, a more robust and straightforward process may be achieved.

Through correct planning, research and implementation a software migration to the cloud may prove to be achieved in a manner which is suitable for the organisation and the software which is being migrated. By following this approach in addition to the recommendations made by cloud providers, it is envisioned that issues would be reduced at the outset of the project and in future iterations of the software (Sabiri *et al.*, 2015).

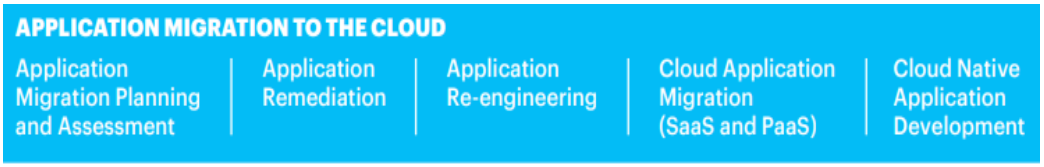


Figure 9 Migration approach ("Accenture Cloud Factory," 2017)

In order for the transition to the cloud to be a success and to have as limited negative impact as possible on the organisation during the migration, there are a number of steps and reviews that are recommended to be carried out at the outset and throughout any cloud migration project. These recommended steps aim to understand the limitations of the existing system while establishing what would be deemed a success once the migration was complete (Mwansa & Mnkandla, 2014).

Step A – Understand existing environment:
This aim of this recommended step is to understand the skillsets, tools and limitations of the existing development environment.

Step B – Determine and evaluate the dependencies and concerns that may arise during the migration process:
This step aims to understand what aspects of the project will impact the migration process and any possible weaknesses of dependant systems.

Step C – Migration Success Factors:
By establishing a number of success points at the beginning of a project and revisiting them throughout, the progress and success of the transition to the cloud can be gauged on an ongoing basis.

7.4 Understanding cloud provider SLAs and dependencies

As a result of the increased uptake in cloud hosting and the push for organisations to move to the cloud, the rate at which change and innovation is occurring is huge. These updates are accelerating the changes that cloud providers need to make and in line with this, they are adapting and tweaking their cloud hosting offerings accordingly. These changes mean increased functionality, performance and overall usability improvements for end users.

This is something that all users who are moving or setting up applications on the cloud need to be aware of. The rapid rate of change means that the underlying systems and configurations of the cloud may need to be updated on a regular basis. What became apparent as part of this study was that users may not be fully aware of the timelines, impact and overheads (in terms of changes needed) aligned with the updates which are rolled out by cloud providers. Issues may begin to occur if users are not fully aware of the dependencies that their software has on the cloud. This means that an unexpected update to a cloud hosted environment may negatively impact the software which is installed on it.

In order for cloud users to fully integrate and be confident with the cloud they need to be fully aware and establish SLAs with cloud providers Toueir *et al.,* (2013). By enabling this clarity, robust testing phases can be built and implemented around scheduled changes, updates and maintenance to cloud hosted environment. This will introduce stringent checks and verification which can be built upon for future iterations of the software.

7.5 Conclusion

In light of the recommendations described above, there are certain changes that users of cloud services could make which would ensure that the transition to the cloud was structured, clear and future proofed. By incorporating a clear vision and processes, issues now and in the future could be mitigated against. These recommendations are particularly relevant to legacy software that is migrating to the cloud because this transition needs to occur in a structured manner so that problems are limited.

CHAPTER 8 CONCLUSION

The aim of this research project was to understand if differences are apparent between software testing practices carried out on locally installed software verses software that has migrated to a cloud hosted environment. In addition, this research aimed to highlight if in fact there were any standardised ways i.e. frameworks or methodologies for migrating software from a local environment to the cloud.

In the preceding chapters of this research project, each of the points highlighted above have been examined in detail. This has been done by investigating existing material, in the form of a literature review as well as gathering new information from experts by carrying out semi structured interviews. The reason for taking this approach was to gather, collate and understand information which originated from many varying viewpoints.

What has come to light following this research is that there are in fact differences in software testing practices when software has been migrated to a cloud hosted environment. It has transpired that the process of migrating software to the cloud may in fact introduce increased software testing scenarios as well as possible benefits to the organisation in the form of cost savings or infrastructure scalability.

It is also important to note that as part of this research project, no standardised or catch all methodology was identified which could be used to assist users who are migrating software to a new cloud environment. It is assumed that each organisation and their specific software is unique however, it may be useful for a methodology to be followed as a guide to all future cloud software migration attempts.

Overall, the findings highlighted in this research project have demonstrated that the area of cloud computing is ever changing and evolving. As a result of this, software testing techniques and migration strategies must continue to update and shift their focus to the cloud.

References

Ashalatha, R., & Agarkhed, J. (2016). Multi tenancy issues in cloud computing for SaaS environment. In *2016 International Conference on Circuit, Power and Computing Technologies* (ICCPCT) (pp. 1–4). https://doi.org/10.1109/ICCPCT.2016.7530261

Barik, R. K., Lenka, R. K., Rao, K. R., & Ghose, D. (2016). Performance analysis of virtual machines and containers in cloud computing. In 2016 *International Conference on Computing, Communication and Automation* (ICCCA) (pp. 1204–1210). https://doi.org/10.1109/CCAA.2016.7813925

Canali, C., Chiaraviglio, L., Lancellotti, R., & Shojafar, M. (2018). Joint Minimization of the Energy Costs From Computing, Data Transmission, and Migrations in Cloud Data Centers. *IEEE Transactions on Green Communications and Networking*, 2(2), 580–595. https://doi.org/10.1109/TGCN.2018.2796613

Carcary, D. M., Doherty, D. E., & Conway, G. (2014). The Adoption of Cloud Computing by Irish SMEs – an Exploratory Study, 17 (1), 12.

Creswell, J. W. (2003). Research design: qualitative, quantitative, and mixed method approaches (2nd ed). Thousand Oaks, Calif: Sage Publications.

Dhanasekaran, S., & Vasudevan, V. (2017). Multiple intelligent agent coordination strategy for categorizing and searching appropriate cloud services. In 2017 IEEE *International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)* (pp. 1–5). https://doi.org/10.1109/ITCOSP.2017.8303133

Gao, Q., Wang, W., Wu, G., Li, X., Wei, J., & Zhong, H. (2013). Migrating Load Testing to the Cloud: A Case Study. *In 2013 IEEE Seventh International Symposium on Service-Oriented System Engineering* (pp. 429–434). https://doi.org/10.1109/SOSE.2013.59

Ladhe, T., Lönn, C. M., Nilsson, A., & Uppström, E. (2015). Platform Design Considerations for Transforming a SaaS Solution to a PaaS Offering. In 2015 *International Conference on Developments of E-Systems Engineering* (DeSE) (pp. 353–358). https://doi.org/10.1109/DeSE.2015.49

Link, S., Jakobs, F., Neer, L., & Abeck, S. (2006). Architecture of and Migration to SOA's Presentation Layer, 12.

Manuja, M., & Manisha. (2014). Moving agile based projects on Cloud (pp. 1392–1397). IEEE. https://doi.org/10.1109/IAdCC.2014.6779530

Migrating Applications to Public Cloud Services: Roadmap for Success. (2018), 38.

Murthy, M. S. N., & Suma, V. (2017). Software testing and its scope in CLOUD: A detailed survey. In 2017 *International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)* (pp. 269–273). https://doi.org/10.1109/ICIMIA.2017.7975617

Mwansa, G., & Mnkandla, E. (2014). Migrating Agile Development into the Cloud Computing Environment. In 2014 IEEE 7th *International Conference on Cloud Computing* (pp. 818–825). https://doi.org/10.1109/CLOUD.2014.113

Paradice, D., Freeman, D., Hao, J., Lee, J., & Hall, D. (2018). A Review of Ethical Issue Considerations in the Information Systems Research Literature. Foundations and Trends® in Information Systems, 2(2), 117–236. https://doi.org/10.1561/2900000012

Parveen, T., & Tilley, S. (2010). When to Migrate Software Testing to the Cloud? In 2010 Third International Conference on Software Testing, Verification, and Validation Workshops (pp. 424–427). https://doi.org/10.1109/ICSTW.2010.77

Pol, K. B. M., & Mengerink, J. (2013). Testing Cloud Services. How to test Saas, Paas and Iaas, 17.

Rowe, F., Brinkley, J., & Tabrizi, N. (2013). Migrating Existing Applications to the Cloud (pp. 68–77). IEEE. https://doi.org/10.1109/CLOUDCOM-ASIA.2013.71

Sabiri, K., Benabbou, F., Moutachaouik, H., & Hain, M. (2015). Towards a cloud migration framework. *In 2015 Third World Conference on Complex Systems (WCCS)* (pp. 1–6). https://doi.org/10.1109/ICoCS.2015.7483315

Saunders, M. N. K., Lewis, P., & Thornhill, A. (2009). *Research methods for business students* (5th ed). New York: Prentice Hall.

Singh, A., Sharma, S., Kumar, S. R., & Yadav, S. A. (2016). Overview of PaaS and SaaS and its application in cloud computing. In 2016 International Conference on Innovation and Challenges in Cyber Security (ICICCS-INBUSH) (pp. 172–176). https://doi.org/10.1109/ICICCS.2016.7542322

Strauch, S., Andrikopoulos, V., Karastoyanova, D., Leymann, F., Nachev, N., & Stäbler, A. (2014). Migrating enterprise applications to the cloud: methodology and evaluation. *International Journal of Big Data Intelligence* 5, 1(3), 127–140.

Taha, A., Manzoor, S., & Suri, N. (2017). SLA-Based Service Selection for Multi-Cloud Environments. In 2017 IEEE International Conference on Edge Computing (EDGE) (pp. 65–72). https://doi.org/10.1109/IEEE.EDGE.2017.17

The Top 5 Cloud-Computing Vendors: #1 Microsoft, #2 Amazon, #3 IBM, #4 Salesforce, #5 SAP. (2017). Retrieved June 13, 2018, from https://www.forbes.com/sites/bobevans1/2017/11/07/the-top-5-cloud-computing-vendors-1-microsoft-2-amazon-3-ibm-4-salesforce-5-sap/#3e2e21b96f2e

Toueir, A., Broisin, J., & Sibilla, M. (2013). A goal-oriented approach for adaptive SLA monitoring: A cloud provider case study. In *2nd IEEE Latin American Conference on Cloud Computing and Communications* (pp. 53–58). https://doi.org/10.1109/LatinCloud.2013.6842223

Varia, J. (2010). Migrating your Existing Applications to the AWS Cloud, 23.

Wang, Y. S., & Yang, C. C. (2014). A best practice of Java-based applications migration with variety of software solutions. In *The 16th Asia-Pacific Network Operations and Management Symposium* (pp. 1–6). https://doi.org/10.1109/APNOMS.2014.6996551

What is multi-tenancy? - Definition from WhatIs.com. (2014). Retrieved May 23, 2018, from https://whatis.techtarget.com/definition/multi-tenancy

What is virtualization? - Definition from WhatIs.com. (2017). Retrieved April 9, 2018, from http://searchservervirtualization.techtarget.com/definition/virtualization#

Zhang, Y. (2018). *Network Function Virtualization: Concepts and Applicability in 5G Networks.* John Wiley & Sons.

Zheng, J., & Du, W. (2014). Toward easy migration of client-server applications to the cloud. In 2014 9th *International Conference on Software Engineering and Applications* (ICSOFT-EA) (pp. 101–108).

Figures

Figure 9: Accenture Cloud Factory. (2017), 1.Retrieved June 10, 2018, from https://www.accenture.com/t20170322T222254Z__w__/us-en/_acnmedia/Accenture/next-gen-4/journey-to-cloud/Accenture-JourneyToCloud-CloudMigrationFactory.pdf

Appendix One - Interview questions

The aim of this interview is to understand further your technical background and your experiences of using cloud hosting services. This interview will specifically focus on your experiences of testing software that has been migrated from a local environment (on premise) to a cloud hosted offering and the tools/technologies used.

Interview questions:

What is your current role?

What specific area do you work in?

What tools/technologies do you specialise in?

Are you aware of cloud computing as a concept?

What are your experiences with it?

Have you used it in your current/past role?

Have you developed or tested software on the cloud?

If so, in what form? SaaS, PaaS, IaaS?

What services did you leverage from the cloud providers? E.g. databases

What provider did you use?

What were the reasons for choosing this provider?

Was it public/private?

What did you use the cloud for? – Data storage etc.

Have you migrated software from a local environment to the cloud?

If so, what were the specifics of the software?

Once migrated, was this software tested?

Did you carry out specific software testing cycles? Regression etc.

Were any issues encountered once the software was migrated?

If so, what types of issues?

Had these issues been encountered prior to the software being migrated?

What were the testing phases you had in place prior to the migration?

Did you follow a step by step migration process (methodology)?

What sort of tools/techniques did you use for testing on the cloud?

Would you recommend migrating software to the cloud?

Do you think that software that is hosted on the cloud can be fully and robustly tested?

Would you recommend migrating software to the cloud?

How would you recommend performing software migration to the cloud, based on your past experiences?

Appendix 2 – Participant A transcript

- What is your current role?

I work in an insurance role; help to manage a system called Polaris which is a rating engine for insurance product across any different type of product.

- What specific area do you work in?

The insurance centre, insure tech side of things. That is where I work at the moment

- What tools/technologies do you specialise in?

We use things like a Polaris Productwriter tool, this is the main rating engine that we use. We have a run time environment, automation to test software and some deployment software which we use on a daily and weekly basis.

- Are you aware of cloud computing as a concept?

Yes, I am aware of the idea behind it, I have done a little bit of work on it, not too much. It is something that will definitely be changing our roles in the future.

- What are your experiences with it?

Yes, small bit of testing around it, not so much on the deployment side but we have been involved in moving some of our software applications from on-site servers to the cloud, and hosting them there.

- Have you used it in your current/past role?

Yes

- Have you developed or tested software on the cloud?

Yes, we have yes so we have our rating engine which we use in the insurance system is a packaged software that we deploy on to a host server which is now on a cloud server and we have a portal that we use to upload various things such as new deployments via and obviously we`ve done a lot of testing on that as and when the initial phase was worked through.

- If so, in what form? SaaS, PaaS, IaaS?

It's a software as a service implementation as far as I am fully aware. That's the set up at the moment.

- What services did you leverage from the cloud providers? E.g. databases

I believe that a number of the security modules were used, obviously to protect the infrastructure and our data and insure it wasn't open for any malicious use.

- RM: This is private insurance information?

Absolutely

- RM: and you found those modules useful, via the providers?

Yes, the pre-packaged stuff was good, it was helpful to understand and see what the benefits of that were for using them.

- What provider did you use?

Microsoft Azure

- RM: Were they easy to get in touch with in the event of issues?

Yeah, I didn't have any contact with them myself but I believe some of the other team did and there was no major issues from what I gather.

- What were the reasons for choosing this provider?

I personally didn't make that decision, I can't say unfortunately.

- RM: It was a higher up decision?

Yes, I'd imagine there was a deal done with someone.

The usual background information was looked at such as stability, support, scalability, size, reproduction of servers & how quick + safely that can be done.

- Was it public/private?

It was a private cloud.

- RM: It was your own implementation and they provided all of the services?

Steve: Yes, that's it.

- What did you use the cloud for? – Data storage etc.

Mainly releasing software, also storing databases of data, data enrichment and enhancement to enhance the data that we gather to provide insurance products.

Have you migrated software from a local environment to the cloud?

Yes, we did yeah.

- RM: How did that work out? Was it worthwhile and was there a lot of work involved?

There was a bit of work involved but not as much as there might have seemed at the outset, it was made relatively simple by the supports put in place. Was it worthwhile? I think we haven't seen any major advantage or disadvantage of it yet but I think knowing the capabilities of it in the future it will certainly be worthwhile.

- If so, what were the specifics of the software?
- RM: It was insurance software for generating rates?

Yes, it is a rating engine for pricing insurance products and maintaining them.

Once migrated, was this software tested?

Yes, absolutely we use some automation tests using software to test both pre and post deployments to the cloud, essentially we weren't putting anything that we wanted to change so the results before and after so the results should be the same, so we worked on that kind of basis of test cycles and test phases so to provide and prove that there was no impact also so that there was some performance testing going on.

- Did you carry out specific software testing cycles? Regression etc.
- RM: Did you do integration testing?

Yeah of course, we did integration testing early on to make sure that the correct calls between the various systems and frontends were happening and that the new URL cloud structure before we got into the actual regression testing and to ensure that there was no impact to the existing process. Various different test phases and sign offs followed that.

- RM: was this the same testing that was carried out when the software was on premise?

It was relatively similar testing. The main difference was that it was hosted on a cloud server versus an in-house server but the mainframe, the inputs and the front ends all remained fairly similar from that side, between them, once the links between them and the new cloud host were put in place.

- Were any issues encountered once the software was migrated?

No serious issues no, there were teething issues here and there with various different response times and calls to different databases and so on. There were general newbie issues, when you get used to new methods of doing things.

- If so, what types of issues?
- RM: No new issues that weren't apparent beforehand? Straightforward?

It did go relatively smoothly, no major issues encountered and certainly ones that were, were easily resolved.

- Had these issues been encountered prior to the software being migrated?

What were the testing phases you had in place prior to the migration?

The exact same testing where at all possible was carried out, it was the same software that we were deploying before and after the cloud and the in house server. So the test packs that we used were the same results expected from them.

- Did you follow a step by step migration process (methodology)?

The process was worked through in a step by step basis to enable the testing in small increments, as and how things were being set up and deployed. We tried to work in a manner that would find issues early and resolve them early so we followed the process that was put in place and best advised to us between Azure and the in house developers who were working on it.

- What sort of tools/techniques did you use for testing on the cloud?

We use JMeter, QTP, UFT, GUI tools that we have designed ourselves. A number different methods of testing along with Excel and VB scripts.

- Would you recommend migrating software to the cloud?

Yeah I think in the long term I think the answer will be yes. This is only short term at the moment so it's hard to say. But I believe if you ask me again in a few months then yeah I would easily be able to say yes.

- Do you think that software that is hosted on the cloud can be fully and robustly tested?
- RM: Any dependencies that are causing any issues?

No, I believe that it can be tested as easily and thoroughly as it can be when held in house or on a local system or server. Otherwise we couldn't have chosen to make that move and take the risk. We would not have left any stone unturned or open to any error in any way to ensure no failures of systems.

- Would you recommend migrating software to the cloud?

Yeah I think the benefits, which we have not seen any of yet! But potentially the benefits far outweigh the old system and methods.

- How would you recommend performing software migration to the cloud, based on your past experiences?

Yeah I think so. I would definitely recommend it. The benefits its gives us far outweigh the negatives of staying on the system we were on before.

- RM: You can see a lot more systems being migrated?

Absolutely, this is a long journey that we are on.

Appendix 3 – Participant B transcript

- What is your current role?

I am working as a software developer at the moment with a big IT company. I have been working for the past 5 years now on technologies like Java, JavaScript, AngularJS, Postgres and open edge tools.

- What specific area do you work in?
- What tools/technologies do you specialise in?
- Are you aware of cloud computing as a concept?

Yes

- What are your experiences with it?

Yes, all of the software development nowadays has an access to cloud because of its scalability opportunities.

- RM: So you have spent a couple of years working with cloud technologies and looking in to them?

Exactly.

- Have you used it in your current/past role?

Yes, we use it all of the time.

- Have you developed or tested software on the cloud?

Yes

- If so, in what form? SaaS, PaaS, IaaS?

We worked primarily with Software as a Service. But then we did actually work on third party tools (the end product) which is like Platform as a Service and we have a team who worked on the Infrastructure as a service as well for deployment purposes.

- What services did you leverage from the cloud providers? E.g. databases

We did depending on the client's requirements actually so we have AWS, Microsoft Azure packages that we can use. Depending on the clients requirements we actually go with certain cloud providers depending on the relationships with the companies.

- RM: You found those useful?

Yes, they are all similar but they good to use.

- RM: Are they tricky to use?

They have their own different technologies, like Amazon or micro services, there will be different ways of using them but the concept is the same.

- What provider did you use?

Azure + AWS

- What were the reasons for choosing this provider?

Because Amazon are one of the biggest providers of cloud services and then you have Azure as well which was used sometimes depending on clients, its dependant on them and if they want to go with Azure or Amazon or some other providers. Because they have their relationship, there are trade relationships with these companies. So they get credits.

- Was it public/private?

Public, it depends actually, not user actually because there may be data that is shared on the cloud. Also dependant on price, public is cheaper compared to the private cloud. So depending on the project and if it is used for internal purposes or if it is strictly for a customer or end users, depending on the different domains that it is in.

- What did you use the cloud for? – Data storage etc.

We used it for scalability options and for data storage or computations for different projects where we need computation abilities for the application or where we have extra servers that we want to add on to.

- Have you migrated software from a local environment to the cloud?

Yes we did, we have project that were taking decades old (20 years +) applications and moving them to the cloud. In those applications, we came across a challenge whereby we have blockages or bottlenecks in software when it comes to handling multiple queries or if there is a change in technologies. So we have to migrate those platforms into the new ones and while we are doing that we are making sure that we are using the cloud because we don't want to be restricted to one location. So we want to have an option for other teams working globally to have access to the clouds and the resources.

- RM: So it is like a share ability kind of thing?

It is kind of giving a shadow around the world so that we use the same platform everywhere with the same kind of strength or latency.

- RM: Were there concurrency issues with this? Was there a portal that you logged in to?

Yes, we did have to do the installation for that e.g. deploying all of the resources, the infrastructure team has done that, we call this, infrastructure as a service. Where they deploy all of the dependencies, all the software, all the resources initially. As a

developer, you just work on it straight away by working on whatever technology you are working on. Just deploy those thing s and start working, we have software version control tools as well that take care of the code database.

- If so, what were the specifics of the software?

Different software depending on the project, sometimes we use micro services which include a payment option, bookings, global bookings or distribution systems or travel booking systems, depending on the client, it is software or applications that are sitting on the cloud and we can access them at any point in time.

- Once migrated, was this software tested?

Yes, there was rigorous testing carried out on that, we have our testing team that works on that as well. As a developer we work on the functionality and sanity checks for them. The rest of the software team work on load testing and scalability testing and making sure it is rigid and dynamic to random requests and is able to handle different scenarios.

- Did you carry out specific software testing cycles? Regression etc.

To an extent yes, but there are other testing like load testing where they actually use 1000% scale to try and break the server and try to hit maximum server amount because we have applications that are going to be used by a lot of people e.g. thousands of people at the same time. You cannot test that in reality, we have a team that actually works on virtual and simulated testing, they have their own testing tools.

- Were any issues encountered once the software was migrated?

Yes, loads of them. A lot of issues were to do with deployments because firstly we have to make sure that the functionality is correct when we do the migration, there is often changes in technologies, some of the old technologies we use are not really compatible with the new technologies or if we have the old technology working on the new browsers like IE/Chrome, they may behave differently so we have to make sure that our new technology works on both ways, i.e. it works with the old system and with the DB as well. When we do the migration, we make sure that it works with the old system at the same time.

- RM: So you have to make sure that it is backwards compatible?

Not 100% but yes, we make sure that it is to some extent.

- If so, what types of issues?

Had these issues been encountered prior to the software being migrated?

There are issues ongoing throughout the cycle of development, we have ongoing development so we are always trying to solve them. Even if it is in a deployed state, we still get tickets that are logged and we try to work on the issues.

- RM: These issues had only become apparent because the software was now on the cloud and the new environment that it was installed on?

I think it was to do with the new environment because the old system doesn't really support these kind of rigorous tests and operations. So the old system, there would be some issues and then straight away we have development issues whereas now, we have accessibility issues because the system is being accessed throughout the world.

- So these issues hadn't really been encountered before the software was migrated?

No, we didn't really keep track of those so I'm not really sure about that.

- What were the testing phases you had in place prior to the migration?
- Did you follow a step by step migration process (methodology)?

In software we have the normal software methodology such as Agile, we try to work on a feedback basis and try to do one step at a time and then test it and then work on the enhancement aspect at the same time as well. Sometimes we have a lean process where we have to do something within a deadline which is mostly the case, so we go with the lean Agile approach.

- RM: So you have a plan as to how you are going to move the software over?

Depending on the length of projects, some projects go with scrum where have a very limited window of time to do development and deployments so in that case we use a scrum methodology as well.

- What sort of tools/techniques did you use for testing on the cloud?

We have internal tools that we use, we don't really use external tools for security reasons. We prefer to use in house tools for testing and making sure that functionality is ok, and we also use sanity checks, which are done by the developers.

- Would you recommend migrating software to the cloud?

I think yes, from one point of view because of scalability. Cloud gives you a scalability option and gives you the benefit of accessing the data anywhere. If you have data in one database sitting somewhere in America and something happens like, there is an earthquake and you don't have access to that information. Cloud gives you the security

of keeping your data safe and safe on the network so it's like in a number of locations and you have a backup as well.

- Do you think that software that is hosted on the cloud can be fully and robustly tested?

Over the period of time it has developed really rigorously so now we have computing which gives an option to programme the nodes and making sure that there is less latency. It has been tested and it is fairly good as compared to the old system and most of the companies use cloud so I think that yes is the answer.

- RM: So in general, you can do everything that you need and the testing is ok. But, would you say that there is more rigorous testing needed for something that is installed on the cloud?

The most important thing when you have data in the cloud is security. So we have security team that are working on a lot of issues and trying to see if everything is ok because there are a lot of ways that data can be accessed. But we just have to make sure that there is no involvement of a third party or somebody else from the network to access the data. So from a security point of view, this is one of the key concerns for all of the companies and I think that, that is very important.

- RM: Is there a lot of testing that goes on around that to make sure that everything is safe & secure?

Yes, to an extent of course.

- Would you recommend migrating software to the cloud?

Yes, I would because it is easy for everybody to access if you have a global company and you don't want to keep something in the same location and by migrating to the cloud everyone will have access and there will be no latency and use the resources and then you can actually keep track of who is resourcing and trying to create tickets and history checks as well.

- RM: And then you can make plans for future deployments etc.?

Yes, and you can scale as well.

- RM: Have your company got more plans to migrate software to the cloud?

I think by this time, we have all of our applications on the cloud anyways. We have hardly any systems that are in house but we have a few internal project that we keep in house.

- How would you recommend performing software migration to the cloud, based on your past experiences?

There was a learning initially, I started in 2010 and then there was a lot of training involved initially to understand the concept and I think that now it has reached a level where it is pretty straightforward and we have different teams working simultaneously.

- It is part of the business now?

Yes, it is.

- Do you have any further insight into the cloud in your experiences?

Yes, I think that the cloud has really emerged and now we obviously given more networking, operations and software development options such as SDN, open daylight all these open resources. These are making the latency reduce because we did have issues that were to do with somebody, somewhere else was not able to access the data at a faster level because of network issues and we have faults in our networks. So, companies are working towards that just to make so that you have cross platform data access, you can use your laptop or iPad or desktop. It gives you more robust and dynamic opportunities.

- RM: In terms of software testing, there were definitely issues at the start when the software was migrated? But, it takes time for them to come through.

Exactly, 10 or 15 years ago there was just a laptop or a computer with the accessibility with limited microprocessor speed. Now we have highly dynamic processors which could be in a phone itself or laptop or iPad so different devices behave differently and has different requirements in terms of contents. So there is rigorous development as well, we just have to make sure that everything is, that the data is slick, accurate and as fast of possible.

Thank you!

Appendix 4 – Participant C transcript

- What is your current role?

So currently I am a QA tester in a reinsurance company here in Dublin.

RM: Ok very good

- What specific area do you work in?

So within the industry it's a reinsurance software. So I suppose I am the main QA tester in the risk side of the company relating to reinsurance and policies like that.

- What tools/technologies do you specialise in?

I am mostly a UI/front end tester, so I do a lot of manual testing and then we are currently expanding our automation portfolio tests so we are kind of upskilling there slowly but surely and trying to make some progress there.

- Are you aware of cloud computing as a concept?

Absolutely, yeah we would use a lot of cloud computing in my current role in my current company and in my previous role as well. We had a lot of products that were based on the cloud.

- RM: So you have good hands on experience (of cloud) in your last couple of roles?

Yeah, I have

- What are your experiences with it?
- Have you used it in your current/past role?
- Have you developed or tested software on the cloud?

Yes, that it most of my job at the moment is testing software that is based on the cloud.

- If so, in what form? SaaS, PaaS, IaaS?

Yes, it was mostly software as a service. This role is reinsurance software and in my previous role it was pharmacy dispensing software.

RM: OK very good

- What services did you leverage from the cloud providers? E.g. databases

Yes, I wouldn’t be specifically involved on that side but I know our data team use Amazon Web Services with RedShift for all of their data. They are after doing a massive migration on to that so that took the last nine months to complete. It’s pretty much complete now so that was a massive migration on our side. I had some minor exposure to testing on that. It wasn’t with any cloud based programs but it was something that the company were delighted to get done.

- What provider did you use?

Yes, it was Amazon web services.

- What were the reasons for choosing this provider?

Well I think they went with most of the cloud because it is very easy to use. It is very adaptable and adjustable. I think they went with this provider because the support would be excellent. They have a very good reputation within the industry. It is very easy to use itself and I would imagine the cost came into it as well. I didn't have anything to do with that but I would think that is something that they definitely had in mind.

- Was it public/private?
- RM: I am guessing that it was probably private because it was probably specific information.

Yes, there was a lot of complex and confidential data involved so it's very private and again Amazon would have a great reputation for being highly secure in this day and age where a lot of leaks etc. happen.

- What did you use the cloud for? – Data storage etc.

On the data team they used it for data storage etc. but we used it for hosting our programs so that they are all hosted on the cloud so that it is not installed on anyone's machine it is just like a web address that you could navigate to. This makes it easier in that we remove all installation processes, it can be upgraded in seconds. It is just much easier to use.

- RM: So it is almost like a URL that you go to and sign in to?

Yes, you just go to the URL and sign in with your user name, password and your permissions and you are good to go. Also, you can it from any computer in the world as long as you have a secure connection.

- Have you migrated software from a local environment to the cloud?

Yes, in my previous role that was exactly what we did so we went from having (in the pharmacy industry). So we went from being a locally installed product that involved a lot of effort to install and maintain and a lot of effort to support as well. It was an online web based application that was literally one web address for any pharmacy to log in to and they had access to all of their information. It was incredibly easy to maintain and support. Very easy to upgrade and adapt. It was a much better, more fluid system. It was something that the users really appreciated. They were a little bit

apprehensive but just with the migration they were delighted to get it. It was something that was a massive improvement.

- RM: So there were big benefits from migrating it from a local install to the cloud?

Yes, absolutely. It allowed the company to expand its market dramatically. They became the first company in the UK to supply cloud based pharmacy software. It was a complete/massive/ unique selling point. It made a lot of money and it was something that they reaped great benefits from.

- If so, what were the specifics of the software?
- Once migrated, was this software tested?

Absolutely yes. It was a massive tests. It was a huge suite of testing that was done over a phased basis. We also transferred users over on a phased basis. There was a massive amount of test and endless regression testing was carried out, sanity testing, smoke testing on new builds. There was system integration testing there was security testing, there was database testing. There was an unending amount of testing. It was incredibly important because there was so much highly secure data, patient data, details from the NHS that were incredibly sensitive. It had to be seriously tested.

- Did you carry out specific software testing cycles? Regression etc.
- RM: Regression, smoke testing, integration?

Yes – Sanity testing, Load testing.

- RM: So it was rigorous? End to end, all different aspects were covered?

Yes, it was.

- Were any issues encountered once the software was migrated?

There were some minor issues at the start which expanded a bit really. We had problems with wiring almost like where some of the ports were incorrectly networked together. It seemed like it was, if they had of gone live then it would have been chaos. SO that was a lot of work to make sure that everything was ok. That was part of the reason that we did so much testing. We had to make sure that all of the messages were being sent and received correctly and that all the security authentication was being sent and received correctly. There was issues at the start because with the ramp up/transition to cloud based software there was an almost immediate rise in the number of customers, which doubled or trebled overnight.

- RM: So the capacity rose hugely?

Exactly. And there was a lot of load issues and stuff like that so, once that was actually levelled out the system was set up for the amount of users that were going to be using it on an hourly/daily basis. It was then a much better/smoother system.

- RM: So it wasn't a straightforward drag + drop exercise?

No, not at all. It was a definite change. It was almost a redesign but it was actually a proper redesign. It was an opportunity to make the product a lot better.

- RM: So the previous version of the software wasn't capable of handling the performance/increased loads?

Not so much the loads, it just couldn't keep up to date and maintain in line with the demands of the customer. Like, the customers were asking for specific functionality and it was very difficult for this. If 10 different customers want the system to do 10 different things, it was very hard for us to program that into our old system. While in the new system, it was very easy to just create a new branch specific to that user and give them their own tailored version of the program. It's much easier to do that than it was with the old system.

- If so, what types of issues?

Had these issues been encountered prior to the software being migrated?

Yes, they were certainly new. It was nearly a side effect of this, we nearly expected them to come up because we expected to come across some issues but they were absolutely massive and incredibly important. So it was really important to get them levelled off pretty quickly.

- What were the testing phases you had in place prior to the migration? Regression, smoke, all of the usual's?

Absolutely, we were quite on top of that. We needed to have/be aware of the importance of all of the testing so it was important to have that levelled off.

- Did you follow a step by step migration process (methodology)?

There was a very detailed step by step process. The project manager would have been over that and I would have played a role on the testing side. So you know, every time we pushed a new piece of software to the cloud, before it went live obviously I would have aggressively tested that with regards to software quality as well as integration and security. That was very important.

- What sort of tools/techniques did you use for testing on the cloud?

As I mentioned, we did some automation testing. A lot of it was manual, just because at that time we didn't have an automation test pack to access because we didn't have the capability or the product built in a way that we could automate it. Or, test it automatically, ow whatever.

- RM: So you weren't using like QTP or JMeter? It was all in house tools?

Exactly, yeah.

- Would you recommend migrating software to the cloud?

Absolutely yeah, it has transformed the way that my team works. It has made their job a lot easier. It is a lot less frustrating, when you are trying to upgrade. You are using the latest technology, you are giving the customer the best tool available. It is much easier to do that when you have the best technologies. When we were working on a locally installed software we were (PCs), it was very hard to upgrade. It was very hard to make sure that users were on the same version. It was very hard to support user queries and questions. So, definitely glad.

- Do you think that software that is hosted on the cloud can be fully and robustly tested?

It depends, I think. Yeah, it does need to be tested heavily, we do need to do as much as we can. We do put a lot of effort in to testing to ensure that software on the cloud is ok. But, I do think that it can be robustly tested, the time and effort just needs to be put in to it.

- RM: So there is a lot of input/overheads required? Because, when it is installed on your local environment you can mess around with it whereas on the cloud it is a bit out of your control?

No, I find that we can have our own QA versions of software. One of the things that I like is that when software is based on the cloud you can instantly create a different branch of code and have that isolated so that you can do as much testing as you want. But I think to robustly test, you need to invest and invest in a lot of different automation tools to save QA time. You need to put a lot of emphasis on just testing and the robustness and that.

- Would you recommend migrating software to the cloud?
- How would you recommend performing software migration to the cloud, based on your past experiences?

I think it needs to be done in a very phased/controlled basis. We were lucky, our project manager was kind of experienced in this and he had a set plan such as, what we were going to do and when. Like, you can't just throw it all up there and hope that it works. You have to literally get the basics up there, like the straw house. Literally, like the least amount of software that you can use to test and you have to expand that a bit more. You have to really, really aggressively test it. A lot of negative, edge case testing. Where a system would perform in one way if it is locally installed whereas if it was cloud based it may not have the support there. It has to be dramatically and very heavily tested to be sure. But generally it is worth doing. It is an investment at the start for sure and it is frustration and it is just hard to do but at the end of the day you are left with a product that is far more scalable. Far more user friendly, way better. Customers will feel the benefit instantly. It is far easier to support and I would imagine it is a lot more cost effective.

- RM: And from your point of view, following a structured methodology and framework was a big help for the project as a whole?

Absolutely, yes.

Appendix 5 – Participant D transcript

- What is your current role?

My current role is a team lead/senior analyst with ABC insurance. We provide general insurance across a variety of motor and home products within ROI and Northern Ireland.

- What specific area do you work in?

At the moment I am a senior test analyst within ABC so we would test on a variety of projects both for external clients and for internal projects. So whether that would be a new website or a new CRM service, or new apps that we want to provide to the clients or small changes with regards to our own internal systems.

- What tools/technologies do you specialise in?

At the moment we do most of our testing via ALM Quality Centre. So we have all of our test cases up there and we run them through that and then we generate all of our reports via that as well. Recently we have been using a new CRM system that we have been rolling out to the agents and to the branches across Ireland at the moment.

- Are you aware of cloud computing as a concept?

Yes, I am. I suppose cloud computing is everywhere essentially. Within ABC, we are only getting in to it now. ABC would be a reasonably old company so they would have used a lot of legacy systems and nearly archaic ways of working. So they are now starting to move into modern technologies.

- RM: So there is a big uptake of them now?

Yes, so we are seeing that in some of the new projects that we have been doing.

- What are your experiences to date with cloud computing?

Across all aspects of life. In my professional career, the most recent to date would be the new CRM project that I was speaking about. This was a project where we were migrating a CRM tool that we had hosted internally and we were migrating that up on to the cloud to store and to host.

- Have you used it in your current/past role?

Yes, we were hosting it via AWS.

- Have you developed or tested software on the cloud?
- You mentioned earlier that it was a SaaS implementation?

Yes, exactly.

- What services did you leverage from the cloud providers? E.g. databases

Yeah, we were using it to host our CRM tool and our CRM platform and once it was there then we began testing it.

- What provider did you use?

What were the reasons for choosing this provider?

As far as I am aware, obviously AWS has a fantastic reputation. One of the reasons it was chosen was due to the support that they provide and their reliability. That was one of the key driving factors behind the choice.

- Was it public/private?

It was private as far as I am aware.

- RM: Because it is insurance/private information?

Yes exactly, and with the GDPR rules for compliance you have to make sure that policy holder information wouldn't be at risk.

- What did you use the cloud for? – Data storage etc.

Have you migrated software from a local environment to the cloud?

- RM: So you mentioned that ABC are doing migrations from a local server to AWS?

Yes, exactly.

- If so, what were the specifics of the software?

The software was essentially a tool that would allow the agents within ABC to be able to deal with the clients or customers in a more all-round way. It would provide a whole 360 view for the agents. In the event they had to interact with any customers they could go to this site and access all of their information in one place.

- Once migrated, was this software tested?

Yes, we did numerous testing cycles both integration testing and verifying that it integrated correctly so that we could access all of our data that we had sent up and all of the components worked correctly.

- Did you carry out specific software testing cycles? Regression etc.

Yes, we would have tested locally so we did some testing on it then to verify that it was all still working correctly. Then we did some performance testing as well to verify that once we were to make it live that it wouldn't be, that the performance would diminish due to the increased usage.

- Were any issues encountered once the software was migrated?

We had a few migration issues as well as environmental issues. Seeing as a lot of the data within ABC was legacy data. That resulted in some integration issues with that data. So we had to investigate them and seeing as the integration issues were flagged fairly early on when a number of components failed to work as they had previously. These were all flagged and eventually rectified.

- If so, what types of issues?
- Had these issues been encountered prior to the software being migrated?

Yes, exactly. In our previous testing cycles we did not come across these issues so once we had migrated it we had found that there was a few discrepancies and issues that appeared.

- What were the testing phases you had in place prior to the migration?
- RM: Was there new types of tests needed because it was on the cloud?

We had to do some increased performance testing and we did more in-depth testing cycles after we had migrated it just to verify that it would be fit for purpose and that it was all working correctly.

- Did you follow a step by step migration process (methodology)?
- RM: Did you just focus on the testing?

There was a methodology in place but from our testing/QA side of things we more focussed on the testing.

- What sort of tools/techniques did you use for testing on the cloud?

Well for the tools we used QC (Quality Centre) and for performance we used Load Balancer. I think we also used that. We had a few data specialists in with us to verify that the data was being hosted correctly as well, that it wasn't being damaged as such.

- Would you recommend migrating software to the cloud?

Yes, I don't see why not. It is both cost effective and from a storage perspective it allows you to host far more than having it locally.

- Do you think that software that is hosted on the cloud can be fully and robustly tested?

I have a feeling this is so. I think once you have a good test strategy in place and the proper discussions are had beforehand.

- RM: So that you are aware of all of the dependencies?

Yes, I think so. Once you are open to them then I think that it is definitely possible.

- How would you recommend migrating software to the cloud, based on your experiences?

Yes, when we were migrating to the cloud we did a larger migration.

- RM: A big bang approach?

Yes, so I would recommend doing an incremental migration. So individual components rather than trying to deploy all of the components at once. I feel if we had done more incremental components it would have resulted in our testing being a bit easier and more focussed and streamlined.

- RM: So that you could focus on an individual module that was migrated?

Exactly. So from our lessons learned that would have been one thing that would have come out of the project.

- RM: And do you think there would have been less issues apparent if you had have done it that way? Such as integration issues?

Yes, I think issues would have been highlighted earlier through doing that and as a knock on effect, rectified issues that we may have had further down the line.

How would you recommend performing software migration to the cloud, based on your past experiences?

Appendix 6 – Participant E transcript

- What is your current role?

Currently a software consultant specifically related to the R&D tax credit but my background is in engineering and junior development roles/junior software development roles.

- What specific area do you work in?
- RM: So your background is in software and you have a good bit of experience there?

Yes, at this stage I am in more of a consultant role at the moment but I have four years' experience in the field of software, before migrating to this role. For the time being.

- What tools/technologies/programming languages do you specialise in?

From a general software perspective, my strengths would be Python/Java and some web technologies such as HTML/CSS, some JavaScript and exposure to cloud computing as well. That is my area. Primarily Python.

- Are you aware of cloud computing as a concept and you know a bit about it?

Yes, I mean it is definitely something that I am really thinking of educating myself on and what it is going it's just a rising field and it is a concept that I am aware of and more so as the months progress because it is getting very big.

- RM: Everyone seems to be jumping on board with it
- What are your experiences with it (the cloud)?

Day to day now not so much but in previous roles I have had exposure such as I used to work in XYZ Ltd and they had some projects around Internet of Things and around implementing it in a manufacturing environment. I have also done some work with the likes of Amazon and AWS and had some exposure to that as well.

- Are your organisation that you worked in now/in the past as a whole involved in cloud computing?
- RM: It is something that they work on a good bit?

As a whole perhaps no, but XYZ Ltd are huge and they provide a lot of the hardware and this is what they are most known for, kind of on the server side .But is something that they are pushing to move into as another branch of the organisation.

- Have you used it in your current/past role?

Previously so as I mentioned, we did a bit of a project in relation to setting up an Internet of Things network on the cloud using manufacturing process tools. Using their

connectivity and the sensors in the manufacturing plant. And some side/personal projects, I have worked with a couple of guys on app development and stuff that have required/decided to move to the cloud.

- RM: Then in terms of college and stuff you have probably had a bit of exposure?

Yes, so I suppose when studying for my diploma yeah it is such a new and emerging field that it is really fed into the college and the course material. It is becoming a core part of it.

- Have you developed or tested software on the cloud?

Yes, so mainly functional testing. So before we had an app called Beezer that I worked on with a guy. It was a side project, a personal project ourselves. We decided to move some of it to the cloud. And some of the side projects in XYZ Ltd yes. I guess it was a cloud based project, from the get go so we had to look at testing it.

- RM: So there is no avoiding the cloud now?

No, no avoiding it. It is a critical part of organisations now so. It is the buzzword of the day.

- If so, in what form? SaaS, PaaS, IaaS?

Yes, from different extents. Even something like DB as a Service, it is all this idea of scalability and not having to have your own hardware I guess. Letting someone else do the maintaining work. Just using it as a utility I suppose. So yes, I am aware of those concepts.

- What services did you leverage from the cloud providers? E.g. databases or any IoT, so any of that kind of information?

Yeah, so in terms of services that I used. So example for storage we were using DynamoDB, we looked into Cognito, it is another tool that Amazon have but just for what we needed e.g. hosting an app so it wasn't really critical or required for what we needed to do.

- What provider did you use? Have you experience with different providers?

So I suppose most recently it was the side project, working on the app development so it would have been AWS or Amazon in general. It's at the forefront and particularly for personal projects it is good because AWS educate is a great resource where you have loads of information and resources there so it is a really good place to learn and get exposure to it and all the rest.

- What were the reasons for choosing this provider?
- RM: Why did you choose AWS?

It is tried and tested, it is accessible in terms of resources. For small scale projects too it is really easy to sign up and it is the first port of call to head to for myself. It's kind of the go to tool. We did consider other guys, the likes of HPE and stuff but, Amazon was a good place to start for us.

- RM: So there was a large range of tools that you could leverage?

Yes, there are a lot of tools/information and resources and reasonable cost as well. That was a big attractor.

- Was it public/private?

We used the public cloud because it is an app that we are trying to get exposure for so it's not going to make millions but it was a public cloud that we were using.

- What did you use the cloud for? – Data storage etc.

As I mentioned before, mainly data storage and for hosting primarily and having that flexibility and scalability so if we wanted to test it we had a couple of hundred users just getting it out there to people that we know. It's a movie and TV review app so we are trying to build it up as a bit of a social thing as well. So we have that bit of flexibility there as well if needed.

- Have you previously carried out migrations internally within your organisation? (I.e. server – server within your organisation?)

Yes, so back to the previous IoT project, in the manufacturing environment so yeah we had some internal migration and as I mentioned, moving some software from a local environment to a cloud and the initial prototype and work was done locally on, onsite hardware but we had to subsequently move it to the cloud when we wanted to push it out to the manufacturing floor and actually test it in the field.

- RM: So you did move it internally and then move it to the cloud?

Yes so it was a bit of a side project in terms of the main thing is you know if a business critical requirement needed our space then we were pushed it around internally then as needed.

- RM: So that was the IoT software that you were working on?

Yes.

- Have you migrated software from a local environment to the cloud?
- If so, what were the specifics of the software?

- Once migrated, was this software tested?

Yes, so as I said the goal of the software was to pull all of this extra data from these process tools that were kind of measuring and taking parameters and mightn't necessarily be used and just stored on the tools itself.

- RM: Like telemetry data?

Yes, telemetry that was kind of sitting on the process tool and going to waste whereas we were trying to pull all of this data and make some use of it and do some processing and add it to the cloud. So we had to carry out testing for sure so we had to take one process tool in our lithography department and we set it up so that the devices were sending telemetry to the cloud and we had to do some integration and stuff in that regard as well. Just initial functional testing as well just to see the kind of capability was and just the initial performance was just for a single tool.

- Did you carry out specific software testing cycles? Regression etc.
- Were any issues encountered once the software was migrated?
- RM: So anything at all that came up?

Yes, one issue encountered when we started initial testing was that these manufacturing tools weren't designed with IoT or cloud computing in mind so when we were trying to pull data off these tools when their main job was to process product so we found that the custom software that was on this mechanical tool working it so that integrating with it and trying to actually pull the data from it was an issue and pull it not just locally but to the cloud as well so we had some issues with that.

- RM: Were there performance issues? Did it work as expected?

No, again, connecting to this purpose built manufacturing tool, the initial simulation/prototype worked well on our local machines up in the lab but when we put it on to the manufacturing floor we started seeing performance issues in terms of the rate at which we could pull data from this tool.

- If so, what types of issues?

Had these issues been encountered prior to the software being migrated?

No, that's it exactly so when we were working with the onsite servers we were initially getting good results but when we migrated to the cloud we had some hiccups like integration issues which affected performance.

- What were the testing phases you had in place prior to the migration?

We didn't get fully up and running, initial testing was based on one processing tool rather than the whole factory and again doing that locally we were getting promising results but the issues were specific to migrating to the cloud.

- Did you follow a step by step migration process (methodology)?
- RM: Any recommendations from Amazon or was it just that you worked it through yourselves?

We worked it through ourselves, it wasn't/initially we weren't scaling up at this stage. We didn't do a divide and conquer route, we kind of just migrated the whole lot to the cloud.

- RM: So you did a drag and drop?

Yes, so it wasn't mission critical at the time so we had that bit of flexibility that we could just get it functioning first and then we could work it out.

- RM: So you could fix it afterwards?

Yes

- What sort of tools/techniques did you use for testing on the cloud?

So testing for this particular project involved running manual scripts so and manual testing to check the functionality. Going back to Beezer, our app. We used, within AWS they have AWS device farmer which allows you to actually test your app and do some automated testing across a range of different devices and connecting different physical devices on the cloud. Instead of buying 100 mobile phones and testing them out, you can use this app to test scenarios. It is very cost effective so and really accessible.

- Would you recommend migrating software to the cloud?

Yes, so it's a case by case basis, it depends on your requirements but certainly in my experience it is a bit of a no brainer. It has its certain pitfalls you have to consider, maybe availability isn't always as great as they quoted. Again you have to look at your own particular requirements but personally speaking yes it's the way to go. Especially for business, in terms of cost and flexibility. If you are a start-up or a small enterprise going out, you don't want to be making huge investments because the technology is going to be out of date in a couple of years so cloud computing really is the future.

Do you think that software that is hosted on the cloud can be fully and robustly tested?

Look, there are a lot of external dependencies so you have to take it case by case and be careful of your own sort of situation. It is always worth your while maybe if you

have that luxury to do some, to do it locally first and do a lot of your initial testing locally before migrating to the cloud. But yes there is, with AWS specifically there is a lot of good tools and services out there like I mentioned, DeviceFarm so it is getting better. It is still something to be aware of.

- How would you recommend performing software migration to the cloud, based on your past experiences?

Yes, so I think my experience just was a drag and drop situation and it worked fine but it was a fairly small project. In my previous organisation, I think it really depends on doing it in stages. So the divide and conquer method may work better, especially with a lot of users. Amazon are all well and good but, they say you can have your system fully operational during a migration but you have to consider your own dependencies and really plan it out so it worked in our case but in a more mission critical project I think you would have to follow a bit more of a rigorous methodology.

- RM: So for small scale stuff it is ok but, a drag and drop. But when there are a lot of dependencies you would need to?

I think you would really need to plan it out a bit more and perhaps do it in stages.

- RM: Modularise the software so? Do it bit by bit

Yes.