**Are you being served: A Framework to manage Cloud outage repair times for Small Medium Enterprises**

**Abstract** – Hosting software applications within a Cloud based infrastructure represents challenges for Small Medium Enterprises (SME’s), due to the variety of ways in which production outages can occur. We consider repair times for outage events in a framework where these downtimes are used to refocus Systems Operations resources. Using an enterprise dataset, we address the question of how outage events are distributed and what relationship these events have with different types of issues that can occur in a cloud data centre. The proposed framework can aid SME’s to maintain a highly available “On-Demand” service infrastructure, with limited resources.

**Introduction**

European Small to medium enterprises (SME’s) have seen significant growth in recent years, a 71% increase in employment (excluding financial sector) was recorded in 2014. Moreover SME’s employed almost 90 million people [1]. As the European economy continues to recover, both businesses and clients are looking for new avenues to drive growth across the EU as a whole.

One way to provide services with a high market reach is through Software as a service (SaaS) model. This cloud-based approach is seen as a shift away from highly complex bespoke solutions, to more focused and cost effective solutions [2]. As customers demand highly effective solutions to solve their business problems, a cloud platform model can help keep pace with these needs. A single delivery platform is used to host multiple software solutions and services.

However an SME’s will face a number of key challenges when embracing a cloud service model, especially in the area of availability and maintainability. Recent work as been conducted to outline these challenges, which include: outage frequency and duration. Almost all SME’s (93%) employ less than 10 people [3], therefore maintaining a reliable service represents their key priority.

In this paper we describe a framework, which the SME can use to best manage their limited pool of resources. The core idea of this framework is for cloud operations teams to focus areas with high outage times (typically areas with high human error) to reduce the overall time from outage to business as usual (BAU). This paper contains a study of software outage data from a large enterprise dataset. Through the study of this outage event data we show which types of outage events take the longest to resolve, why having standardised homogeneous data centres are key to reducing outage times, and how application types play a role in the duration of resolution of outages.

For Platform as a Service (PaaS) providers, where mutli-tennacy solutions are available, high-level outage data could be shared between organisations to triangulate cross application outage events.

The rest of the paper is structured in five Sections; Section II gives some description of study background and related works. Section III describes the enterprise dataset. Section IV discusses the analysis and method and it is followed by section V that explains the result. Finally, the conclusion and future work is described in Section VI.

**Background and Related research**

*A. Software as a Service (Saas)*

*Talk about Cloud architecture and how it’s used to drive blah blah blah*

*B. Cloud Outages*

A cloud outage is the amount of time that a cloud service is unavailable to the customer. While the benefits of a cloud systems are well known, a key disadvantage is that when a cloud environment becomes unavailable in can take a significant amount of time to diagnose and resolve the problem. During this outage time the platform can be unavailable for all customers.

One of the first cloud outages to make the headlines in recent times was the Amazon outage in April 2011. In summary the amazon cloud experienced an outage that lasted 47 hours, the root cause of the issue was a configuration change as part of a network upgrade. While this issue would be damaging enough for Amazon alone, a number of consumers of Amazon’s cloud platform (Reddit, Foresquare) also suffered the same downtime. [Ref]

While great improvements have been made in relation to redundancy, disaster recovery and the ring fencing of key critical services, the big players in cloud commuting not immune to some form of outage. As of mid 2015 a number of high profile outages were catalogued by CRN website. [Ref] A summary table is included below:

|  |  |  |
| --- | --- | --- |
| **Company** | **Outage Time** | **Outage Details** |
| Verizon | 40 hours | Scheduled maintenance to improve overall reliability. |
| Apple iCloud | 12 hours | A DNS error meant that users were unable to make purchases. |
| Apple iCloud | 7 hours | iCloud unavailable / poor performance affected some 200million users. |
| Windows Azure | 2 hours | A network infrastructure outage resulted in loss of service for all central US users. |
| Starbucks | Several hours | Schedule maintenance resulted in tilling system going off-line. |

*C. Other related Studies*

A number of studies have been conducted in relation to cloud outages and the time observed to resolve problems in repairable systems.

Yuan et al. [ref] performed a comprehensive study of distributed system failures. Their study found that almost all failures could be reproduced on reduced node architecture and that performing tests on the error handling code could have prevented the majority of failures. They conclude their study by discussing the efficacy of their own static code check as a way to easily check error-handling routines.

Hagen et al. [ref] conducted a study into the root cause of the Amazon cloud outage on April 21st 2011. They found that an IT change to route traffic from one router to another while a network upgrade was conducted. The backup router did not have sufficient capacity to handle the required load. They developed a verification technique to detect change conflicts and safety constraints, within a network infrastructure prior to execution.

Li et al [ref] conducted a systematic survey of public Cloud outage evetns.

Kleyner and O'Connor [ref] propose an important thesis regarding reliability engineering. While emphasis is placed on measuring reliability for both mechanical and electrical/electronic systems, the authors do broaden their scope to discuss reliability of computer software. One aspect of interest is their discussion of the log normal distribution and its application in modelling for system reliability with wear out characteristics and for modelling the repair times of a maintained systems.

Almog [ref] analysed repair data from twenty maintainable systems electronic systems to validate whether either lognormal or exponential distribution would be a suitable candidate distribution to model repair times. His results showed that in 67% of datasets the log normal distribution was a suitable fit, while the exponential was unsuitable in 62% all of datasets.

Another random study [ref]

**Data set**

Cloud outage studies have been shown to provide an effective way to highlight the distribution of failure events. These studies can be leveraged by enterprises to pre-empt common failure patterns.

The study presented in this paper examines approximately 250 field outage events from a large cloud based system. The data was collected over a 12-month period (Jan - Dec) and is comprised of four main components: E-mail, Collaboration, Social and Business Support System (BSS). Additionally the failure events have been categorised into the following main categories: Configuration error, Hardware failure, Network and Software defect. The systems have been deployed within three data centres and are used by customers globally. The software is developed in Java and runs on Linux. Product development follows a Continuous delivery (CD) model whereby small amounts of functionality are released to the public on a monthly basis. For each outage event we have access to the full outage report, but we particularly focus on the time taken to resolve the outage with additional focus on the software component and the type of error, which was the root cause of the outage. The following terminology will now be defined to provide clear context. These definitions are given in the glossary of IEEE Standards Collection in Software Engineering [ref].

* Outage Event
* Maintenance window
* Time to detection (TTD)
* Time to resolution (TTR)

This study aims to answer a number of questions. First, How are the times of cloud outage events distributed? Second, does the distribution vary by component? Third, does the distribution differ by failure category? Fourth, does the relationship differ by data centre? Finally what is the relationship between time to detection and time to resolution? In order to answer these four questions, this study is broken down into the following attributes: outage distribution, outage component, outage failure category, data centre location and TTD Vs TTR.

1. Outage Distribution

Probability distributions are used in statistics to assign a probability or likelihood of an event-taking place. In the case of cloud outage events, by analysing the distribution of all events, it may be possible to fit a known distribution to our dataset. If a distribution can be fitted these distribution properties can be used to infer the most likely outcome of an outage event. For example a probability distribution could be used to infer the likelihood of an outage event taking a specific period of time to resolve. Outages distributions are plotted at an overall, component and type level.

1. Outage Component

Recognising the location of an outage event at a component level gives an understanding of a) which components are more likely to contribute to an outage event and b) the relative duration to detect and resolve an outage with respect to a component. For example operations teams may have various probes to determine if an event is likely to cause a failure. Development and test teams may have a suite of test cases to find a certain class of issue. Outage events can provide operations teams with an understanding of potential gaps in their probes and monitoring solutions. Likewise for development and test teams outage events can provide both teams with either weaknesses in feature implementation and gaps in test coverage. Depending on the nature of these test gaps and the size of the test organisation, they may be difficult to close. For this study we categorised our software components as follows: BSS, collaboration, e-mail and social.

1. Outage Failure Types

Over the course of our study, we found a variety of outage events. To give clarity to these different types of outage event, we divided the outages into four main categories: Configuration error, Hardware failure, Network and Software defect.

A configuration error typically relates to where a configuration setting has been set in error. As a result the parent component may operate below normal specifications. Configuration settings may also cause unwanted side effects to sub-components, which consume the parent component.

A Hardware failure relates to a class of problem, which causes a piece of hardware to fail. These failures relate to a malfunction within the electronic circuits or electromechanical components (disks, tapes) of a computer system. Recovery from a hardware failure requires repair or replacement of the offending part.

A network error relates to a class of failure outside of misconfiguration or a hardware failure within the network infrastructure. Network failures can typically present themselves as intermittent temporal network outages or as high latency / packet loss conditions. As cloud data centres contain a number of distributed systems, having a reliable network infrastructure is highly desirable.

A software defect refers to a class of issue, which is triggered through normal operations on the underlying server component code by the customer. These issues are triggered due to the inability of the code to handle either concurrent or parallel usage. Software defects may include issues related to contention under load (e.g. memory leaks, high Disk I/O, CPU usage), concurrency (e.g. deadlocks) or miscellaneous error conditions.

1. Data Centre Location

Understanding the measure of outage events at a data centre level can highlight whether a specific data centre is a factor in the duration and distribution of outage events raised. There are three data centres in our dataset: data centre A (High usage), data centre B (Low usage) and data centre C (Medium usage). Having a correlation between outage duration can be a useful data point for cloud operations teams.

1. TTD Vs TTR

Examining the ratio between the times taken to detect an outage compared to the time taken to resolve an outage is important. Studying detection times can highlight whether firstly an operations team has sufficient monitoring capabilities and/or whether the server side components produces sufficient error events to service the problem. Additionally studying resolution times can provide insight into whether staffing and/or crisis operations processes are sufficient in the case of multiple concurrent outage events.

1. Limitations of dataset

The Dataset has a number of practical limitations, which are now discussed. While the outage event tracking system allows for a granular categorisation system, whereby outages can be mapped to a subcomponent, there are a number of outages, which due to their severe nature can affect more than one component and subsystem. The authors reviewed the functional location of each defect to ensure precision across the analysis of the dataset.

The outage events that form part of this study are from an enterprise cloud system. The outage events are applicable to the software domain of BSS, Collaboration, Email and Social. Additionally the outage events are applicable to the field of hardware failure, software defect, Network errors and misconfiguration of server components.

**Results**

We now explore the attributes of the outage events observed.

1. Outage Distribution
2. Outage Component
3. Outage Type
4. Data Centre Location
5. TTD Vs TTR

**Discussion**

Section IV provided an outline of outage events that were studied as part of our overall dataset, including distribution, component, type, data centre location and a comparison between TTD and TTR. The following section provides deeper analysis of the results. In each section references will be made to each research question asked in section III.

1. Outage Distribution
2. Outage Component
3. Outage Type
4. Data Centre Location
5. TTD Vs TTR

**Conclusion**

**References**

**Intro**

The Adoption of Cloud Computing by Irish SMEs – an Exploratory Study

<http://www.ejise.com/issue/download.html?idArticle=933>

# Cloud predictions to watch for in 2016

<http://searchcloudcomputing.techtarget.com/feature/Cloud-predictions-to-watch-for-in-2016>

## Annual report on European SMEs

http://ec.europa.eu/growth/smes/business-friendly-environment/performance-review/index\_en.htm

# Cloud computing - statistics on the use by enterprises

http://ec.europa.eu/eurostat/statistics-explained/index.php/Cloud\_computing\_-\_statistics\_on\_the\_use\_by\_enterprises#Further\_Eurostat\_information

**Background research**

**Cloud Architecture**

From Google to Amazon - the rise of the cloud catalog

**Cloud Outages**

**http://www.crn.com/slide-shows/cloud/300077635/the-10-biggest-cloud-outages-of-2015-so-far.htm/pgno/0/1**

http://www.datacenterdynamics.com/security-risk/from-google-to-amazon-the-rise-of-the-cloud-catalog/83949.article

10 worse cloud outages

http://www.infoworld.com/article/2622201/cloud-computing/the-10-worst-cloud-outages--and-what-we-can-learn-from-them-.html

**Other Related studies**

### [A study of the application of the lognormal distribution to corrective maintenance repair time](http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix=html&identifier=ADA072320)

<http://www.dtic.mil/dtic/tr/fulltext/u2/a072320.pdf>

A Study of the Application of the Lognormal and Gamma Distributions to Corrective Maintenance Repair Time Data.

http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA124632

Log Normal Distribution in Reliability studies:

<https://en.wikipedia.org/wiki/Log-normal_distribution>

O'Connor, Patrick; Kleyner, Andre (2011). Practical Reliability Engineering. John Wiley & Sons. p. 35. [*ISBN*](https://en.wikipedia.org/wiki/International_Standard_Book_Number) [*978-0-470-97982-2*](https://en.wikipedia.org/wiki/Special:BookSources/978-0-470-97982-2).

Log Normal Distribution

http://reliawiki.org/index.php/The\_Lognormal\_Distribution

To big to fail

<http://cacm.acm.org/magazines/2015/2/182653-too-big-to-fail/fulltext>

**Datset**

Probability distribution

http://smallbusiness.chron.com/role-probability-distribution-business-management-26268.html

Hardware failure

http://encyclopedia2.thefreedictionary.com/hardware+failure