Characterizing Cloud Computing Hardware Reliability

Kashi Venkatesh Vishwanath and Nachiappan Nagappan
Microsoft Research
One Microsoft Way, Redmond WA 98052
{kashi.vishwanath,nachin}@microsoft.com

ABSTRACT

Modern day datacenters host hundreds of thousands of servers that coordinate tasks in order to deliver highly available cloud computing services. These servers consist of multiple hard disks, memory modules, network cards, processors etc., each of which while carefully engineered are capable of failing. While the probability of seeing any such failure in the lifetime (typically 3-5 years in industry) of a server can be somewhat small, these numbers get magnified across all devices hosted in a datacenter. At such a large scale, hardware component failure is the norm rather than an exception.

Hardware failure can lead to a degradation in performance to end-users and can result in losses to the business. A sound understanding of the numbers as well as the causes behind these failures helps improve operational experience by not only allowing us to be better equipped to tolerate failures but also to bring down the hardware cost through engineering, directly leading to a saving for the company. To the best of our knowledge, this paper is the first attempt to study server failures and hardware repairs for large datacenters. We present a detailed analysis of failure characteristics as well as a preliminary analysis on failure predictors. We hope that the results presented in this paper will serve as motivation to foster further research in this area.

ACM Categories & Subject Descriptors: C.4 [Performance of systems]: Reliability, availability, and serviceability
General Terms: Measurement, Reliability
Keywords: datacenter, failures

1. INTRODUCTION

Modern day datacenters (DC) host hundreds of thousands of servers [3] networked via hundreds of switches/routers that communicate with each other to coordinate tasks in order to deliver highly available cloud computing services.

Unfortunately, due to economic pressures the infrastructure that these services run on are built from commodity components [8]. As a result, the hardware is exposed to a scale and conditions that it was not orignially designed for. The servers consist of multiple hard disks, memory modules, network cards, processors etc., each of which while carefully engineered are capable of failing. While the probability of seeing any such event in the lifetime (typically 3-5 years in industry) of a server can be somewhat small, across all machines [1] hosted in the datacenter, the number of components that could fail at any given instant is daunting. At such a large scale, hardware component failure is the norm rather than an exception [4].

Hardware failure can lead to a degradation in performance to end-users due to service unavailability [6] and can result in losses to the business, both in immediate revenue [20] as well as long-term reputation [16]. The first impact of this is that it puts an increased onus on the software stack via added complexity for dealing with frequent hardware failures [14]. Even without regard to the increases in complexity of software [9], diagnosing and servicing these faults, deemed important to DC operation increases the operational expenditure (OPEX) [4]. A sound understanding of the number of failures as well as the causes behind them helps improve operational experience by not only allowing us to be better equipped to tolerate failures but also bring down the hardware cost through engineering. Further, if we develop a model that allows us to proactively predict failures, this can lead to moving workload and data off of such a server in time to avoid any possible service disruption.

Consider an alternate model of building datacenters by packing servers in a serviceless module, e.g., a container [19]. As these carefully engineered modules would contain redundant parts to cope with hardware failure it is imperative to know relatively accurate failure characteristics to avoid overprovisioning. Thus, we would not only want to know the reliability of individual components in order to lower the cost of running cloud computing infrastructures, but we would, in fact, like to evolve a composable reliability model so that we can use it to better design future infrastructures. Such a hierarchical reliability model would help us analyse the impact of whole DC failures, individual rack or container/pod failures, server failures, networking equipment failure as well as individual component failure. This paper focuses on one part of the puzzle, understanding server failures.

The failures that we observe are a complex function of a large number of variables, viz., manufacturing process, deployment environment conditions, workload etc., analogous to a random experiment. In this paper we aim to establish sound observations to the outcome of such an experiment. In doing so we build upon recent large scale studies on hard disk [15, 17] and memory module failure characteristics [18]. While these recent efforts have focussed on detailed analysis of component failures, in this paper we wish to tie together component failure patterns to arrive at server failure

SoCC'10, June 10–11, 2010, Indianapolis, Indiana, USA.
Copyright 2010 ACM 978-1-4503-0036-0/10/06 ...$10.00.

[1] we use the terms machine and server interchangeably in this paper

rates for datacenters. As a first step this can be used by academics to model a large number of design solutions [19, 7, 12, 1]. In addition this is the first step to begin reasoning about the causes behind these observations. While we also present a preliminary analysis on predicting failures, the main aim of this paper is to characterize the faults seen in large cloud computing infrastructures. We make the following four important contributions towards that goal.

- This paper is the first attempt to characterize server failures for large data centers. We present a detailed analysis of failure characteristics and explore the relationship between the failures and a large number of factors, for instance, age of the machine, the number of hard disks it has, etc.

- This is the first work to quantify the relationship between successive failures on the same machine. We find that the empirical data fits an inverse function with high significance.

- We perform the first predictive exploration in a datacenter to mine for factors that explain the reason behind failures. We find, for instance, that the datacenter where a server is located is a great indicator of failures and so is the manufacturer.

- We show empirically that the reliability of machines that have already seen a hardware failure in the past is completely different than those of servers that have not seen any such event.

Section 2 describes how we gather and analyse the datasets used in this study. We begin (Section 3) by describing how we detect hardware faults along with the high-level characteristics of the hardware failures and the associated cost of servicing them. We then present a preliminary analysis (Section 4) of various prediction techniques to explain observed failure characteristics in our datacenters. We then follow it up with a description of related work (Section 5) before concluding (Section 6).

2. DATACENTER CHARACTERIZATION

We begin by describing the nature of hardware found in datacenters. While we have made no attempt to handpick a particular flavor of machines either by composition or by use, we have no means of either verifying or guaranteeing that the characteristics described next are either typical or representative of datacenters elsewhere. Such studies while important, can only be done by either collaborating with hardware vendors (something we are exploring in the future) or by subjecting identical workloads to varying kinds of hardware (something we do not practice currently).

2.1 Data Sources and Problem Scope

Ideally we would have access to detailed logs corresponding to every hardware repair incident during the lifetime of the servers. We would also know when the servers were comissioned and decomissioned. However, without the proven need for such detailed logging no such database exists. Thus, in the absence we resort to combining multiple data sources to glean as much information as we can.

The data sources used in this study were originally put in place with a separate goal and were not necessarily aimed at precise tracking of each of the quantities we are interested in. As a result there is no single piece of database that tracks multiple quantities of interest, i.e., the inventory of machines and their unique ids, the composition of machines (number of hard disks, memory modules etc.), trouble tickets, hardware repairs, temperature and other environmental metrics, performance numbers for the server including cpu, memory load etc. One of the main challenges we face is to combine together disparate data sources that were originally meant for a different purpose and naturally had varying levels of importance to detailed and accurate logging of the fields that is of most interest to us. Furthermore, with organizational changes it becomes difficult to track the original owner of the data source in case of discrepancies. In particular there are three sources of data that is of interest to us and we describe them next.

The first piece of data is the inventory of machines. This contains a variety of information regarding the servers that are used for cloud computing. This includes a unique serial number to identify a server, date when an operating system was installed on the server, the datacenter where this server is located and what role the machine is commissioned for. There are over 50 fields and that gives us rich statistical variables to mine for and we describe this later in the paper. We use the serial number as a unique identifier for the machine id.

The next piece of information that is critical is the hardware replacements that take place. This data is maintained separately. This is part of the trouble tickets that are filed for hardware incidents. Each ticket has a variety of other information including the date when a fault was recorded and ticket was filed, when was it serviced and what server does that correspond to. It also has information on how the fault was fixed i.e. replacing a disk etc. While there are a variety of fields present, a few key fields are missing too. For instance, if a hard disk was replaced in a RAID-6 array, there is no information maintained on which of the 6 hard-disks was replaced. Often details about the hard-disk itself are not maintained, for instance, SATA vs. SAS. Furthermore, we do not know if replacing the hardware actually fixes the problem. It is possible that a failure elsewhere in the system was raising false alarms that lead us to replacing the wrong piece of hardware. In this paper we are, however, interested in first establishing quantitatively the hardware replacements for our cloud computing infrastructure. In ongoing work we are developing models to understand how multiple hardware replacements may in fact correspond to a common ”fault”.

The reader may note that owing to possible human error in recording failure events, we may underrepresent the total set of hardware replacements in our infrastructure. Thus any corresponding cost estimate provided in this paper will be a lower bound on the actual cost. Similarly, replacing a hardware component does not guarantee that it is indeed a hardware failure. We rely on detailed diagnostics to declare whether a hardware component is no longer fit to be under deployment in our servers. Thus, we use the words failure and repair event interchangeably, similar to other researchers [17].

Using the above piece of data we can calculate the total number of hardware replacements seen in a given time on a given server. We can also understand patterns of failure/repair events and the relative frequency across different components. However, these two data sources are still not enough to understand the failure *rate* of individual components. In order to calculate that, ideally we would like to track the life span of individual components i.e., hard disks, memory modules etc. However, that is not available and we need an alternate method. We can get information about the configuration of machines, for instance, the number of hard-disks, memory modules etc. from a third data source. In this data source each component (e.g., hard disk) has a serial-id and an associated server-id where it is placed. We use this to loosely understand the configuration of each machine. Thus, at this point we know how to combine detailed information on the server, its configuration, and the hardware events that are recorded over time.

There are numerous challenges with the coverage of machines and the accuracy of the data. For instance, there are repeated serial

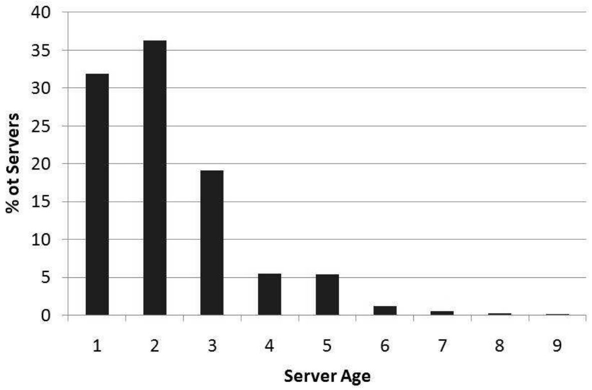

Figure 1: Age profile of servers.

numbers, etc. that are mostly due to human error that cannot be avoided. However, we try our best to present data that is heavily sanitized by careful human intervention and interpretation of the raw data. In doing so we restrict out dataset to a smaller but more consistent size and report our findings below. Finally, this data is only available for a 14 month period. In the future we are looking to extend this dataset. However, to the best of our knowledge, this is the first study that looks at such a large number of servers in production use and tries to understand the reliability of servers.

Our data does not contain sufficient information to help us understand single points of failures for entire racks or datacenters. In this study, we focus on server reliability and the factors that affect it. In an ongoing effort we are building a whole datacenter reliability model. Every repair of a hardware incident contribute to the OPEX, thus, understanding server reliability is important. However, that in itself is only meaningful for current datacenter operations. What if we are interested in cloud computing infrastructures of the future and as part of that are trying to decide how to build servers? We would like to know what components to pick to build that. Thus, we would like to understand the failure patterns for each component. Of course, deriving this from an existing operational infrastructure implies that we are ignoring any dependencies. Nevertheless, in this work our aim, as mentioned earlier, is to arrive at a first cut at the repair numbers and in ongoing and subsequent efforts to mine the root causes within.

2.2 Server Inventory

Here we describe the configuration and nature of machines used in the dataset. The description includes various factors about the machines, including their age profile, configuration etc. In order to maintain privacy, data is normalized wherever appropriate. The goal of this section is to present enough details to allow for a scientific comparison of analogous results against other datasets that the reader may have access to. We also hope that it will serve as a model to base academic studies on.

- **Subset of machines.**
  We have details on part replacement for over 100,000 servers (exact number withheld). This includes details, for instance, when a hard disk was issued a ticket for replacement, and when was it replaced and the details of the server corresponding to it. The collection of servers span multiple datacenters in different countries (and continents).

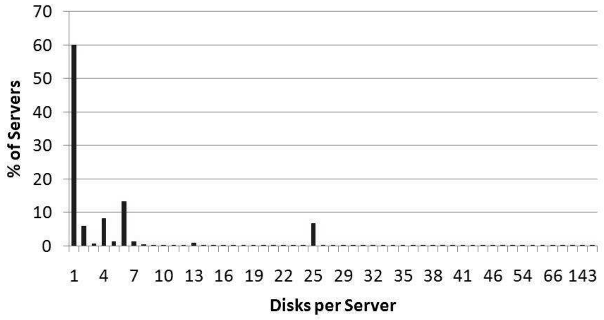

Figure 2: Disk profile of servers.

- **Age profile of machines.**
  The age of the machine when a fault/repair happened is of interest to us. The age can be calculated on any given day and we report the age of the server (in years) at the beginning of our database period as shown in Figure 1. The X-axis shows the age in years and for each such age group, the Y-axis shows the percentage of servers that fall into that category. Around 90% of the machines are less than 4 years old. This is in accordance with the company policy of retiring machines at the end of 3 years. However, we do find machines that are, for instance, 9 years old [2].

- **Machine configuration.**
  We next describe the composition of these servers. On an average there are 4 disks per server. However, there is a huge variation as shown in Figure 2. 60% of the servers have only 1 disk. However, 20% of the servers have more than 4 disks each. We also calculated the profile of number of memory modules in a server and the results are shown in Figure 3. Average number of memory modules per server is around 5. As we can see, a majority of servers have 4 modules. But there are servers with over 16 modules too.

3. CHARACTERIZING FAULTS

As mentioned earlier, there is no easy way to know when a fault occured. However, we do track when a repair event takes place and a ticket is filed. Since tracking of tickets and sending personnel/administrators to fix the fault contributes to the OPEX it can be used as a good substitute. Please note, as mentioned earlier, that in the current work we are interesting in quantifying the number and types of hardware repair events. Arriving at detailed explanation behind the cause of these failures is part of an ongoing effort. Thus, a faulty RAID controller, that manifests itself as multiple hard disk faults will be counted as multiple hard disk failures in our current work, where the focus is on understanding how much money we spent in repairing hardware faults.

3.1 Identifying Failures

We begin by describing the total set of repair events that we saw in the 14 month period across our set of machines. To preserve data

[2] Due to regular purchase as well as retirement of servers, the graph does not look much different when computed at the middle as well as end of the database period.

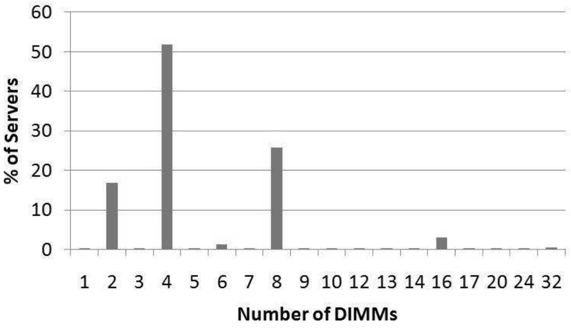

Figure 3: Memory module profile of servers.

privacy, all numbers reported from henceforth, will be normalized to 100 servers. We next describe the number of hardware faults, the servers they were incident on etc., all after applying the same scale-down factor, i.e., by scaling down the reported values as a percentage (i.e., over 100).

We saw a total of 20 replacements in the 14 month period. However, these replacements were all contained in around 9 machines. 9 machine failures in 14 months translates to an annual failure rate (AFR) of around 8%. Let us now consider only the subset of machines that saw at least 1 repair. The average number of repairs seen by a repaired machine is 2 (not shown here). Thus, 92% of the machines do not see any repair event but for the remaining machines (8%) the average number of repairs per machine is 2. Figure 4 shows the distribution of the percentage of servers against a given number of repairs. Thus, over 50% of the repaired servers see exactly 1 repair. The ”knee” of the curve is around 3, thus, 85% of the repaired servers see less than 4 repairs.

3.1.1 *Cost of these faults*

Here we examine the cost of machine failures and hardware repairs. The first cost is the associated downtime of the machines. In addition it costs the IT ticketing system to send a technician to the fault site to perform a repair operation. Finally, hardware repairs cost in terms of the hardware component being repaired/replaced. Assuming the same numbers that Google reported [4] where each repair costs $100 for the technician’s time and 10% of the server cost ($2,000) we arrive at a repair cost of $300. Given an AFR of 8% this amounts to close to 2.5 million dollars for 100,000 servers. It is important to know the relative reliability of different available choices to order from a catalogue. For instance, at $300 per repair, in 6 repairs the cost of repairs is already close to that of purchasing new hardware. Such considerations require a sound understanding of the failure characteristics, the focus of this paper.

3.2 Classifying Failures

Of these replacements/faults, a majority (78%) were for hard disks, followed by a few (5%) due to raid controller and even fewer (3%) due to memory. However, 13% of all replacements came from a collection of all other components with no single component dominating. Thus, we can cut down substantially on the number of failures by improving the reliability of hard disks [3]. However, once we do that there is no single component that dominates in failure rate.

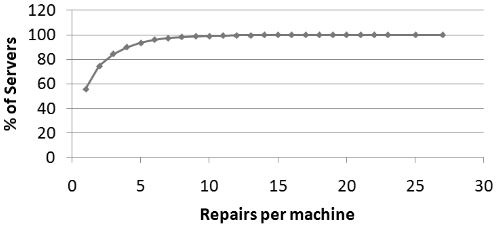

Figure 4: Distribution of repairs per server.

The above analysis calculates the frequently failed components and the cause for hardware repair. This analysis is helpful in calculating the money spent in repairing hardware as well as approximating availability by calculating the associated downtime etc. Let us revisit the serviceless datacenter model once again. In this model, as soon as any hardware component in a server fails, we declare the server as ”dead”. In order to understand the resulting reliability of the server with this assumption, we need to look at existing data in a new fashion. Instead of counting all hardware fault belonging to a machine we now only look for the first hardware fault that is incident on a server. We use this to understand what component triggers the first hardware fault on a server. An analysis yields the following results. 70% of all server failures is due to hard disks, 6% due to RAID controller and 5% due to memory and the rest (18%) due to other factors.

Thus, hard disks are the not only the most replaced component, they are also the most dominant reason behind server failure.

If we look at the total number of components of each type, i.e. disk, memory, RAID etc. and look at the total number of failure of the corresponding type, we can get an estimate of the component failure rate. Using this technique we arrive at the following figures. 2.7% of all disks are replaced each year. This number is just 0.7% for raid controllers and only 0.1% for memory, If we consider all other components in an aggregate ”other” category, then the failure rate for those components is 2.4%. Note however, that this is just an approximation. We do not know, for instance, that which of the many different hard disks in a RAID array fail. We arrive at these numbers by dividing the total number of replacements with the total number of components. This can result in double counting disks in a RAID array. Thus, the values reported here are an upper bound on individual component failure rate. If multiple repair events happen to be for the same disk ("logical") then the actual component failure rate will be lower than what we observe above. However, please note that this would still lead us to a correct calculation of the administrative overhead in repairing these machines [4].

Given that hard disks are the number one failing component we decided to investigate its failure characteristics further. We looked at clusters [5] of servers. For each of those, we calculated the total number of servers that saw at least one disk related repair event in

[3] Assuming independent failures in different components.

[4] Assuming the current service model. Results will be different if we consider a serviceless DC model [19]

[5] A cluster here refers to a logical set of servers put to similar tasks/workloads but not necessarily similar in configuration or even geographical location.

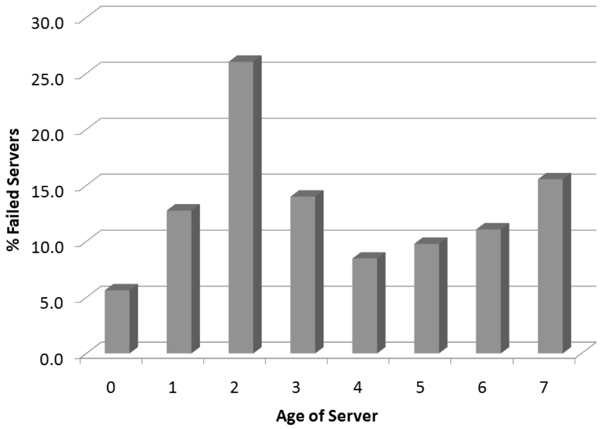

Figure 5: Age distribution of hard disk failures.

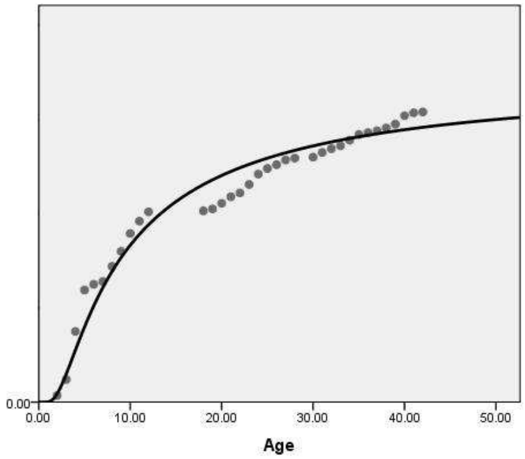

Figure 6: Number of repairs against age in weeks.

the 14 month period. This percentage was significantly high for the top 5 properties (i.e. largest sized clusters), between 11% and 25% of all servers see at least 1 disk event in the given period (not shown here). Note that this is higher than the aggregate failure rate we saw earlier, i.e. 8%.

We calculate the age at which a disk failure occured and the aggregate results are shown as a histogram in Figure 5. Here the X-axis shows the age of the server (in years) when a hard disk fails and the Y-axis shows the percentage of servers that see a hardware failure at a given age. Thus, there are very few (5%) young servers (<1 years) that see any disk failure. The number jumps slightly higher (12%) as the machines are slightly old, i.e., 1 year old. Finally, 25% of the machines that see a hard-disk fail are at least 2 years old. Please note, however, that due to lack of sufficiently detailed data, we cannot use this figure to calculate the failure rate at different ages of a hard disk. In the next section we will use an alternate technique to closely monitor all hardware events occuring on a server.

3.3 Young Servers

One limitation of our dataset is that it is a 14 month slice in time of hardware repair events as opposed to the entire lifetime of the servers. Thus, there is no direct way of knowing all the repairs that happened to the machines prior to day 1 of our dataset. Similarly, we do not know the fate of the servers beyond the 14 month window we are observing. One approach is to do a detailed and careful modeling exercise to understand the failure trends beyond the 14 month period. Owing to the inherent inaccuracies that might introduce in addition to the complexity in the first place we suggest an alternate mechanism. We focus on those machines that have been brought online or put into production during the 14 month period. This ensures that we will be able to track all hardware repairs on these machines.

We show the cumulative number of failures that the servers see as a function of age (in weeks) in Figure 6. The Y-axis has been omitted to maintain data privacy. As can be seen, the S-curve is a great fit (the R^2 value for the fit was 0.973). R^2 is a measure of variance in the dependent variable that is accounted for by the model built using the predictors [13]. In other words, R^2 is a measure of the fit for the given data set. (It cannot be interpreted as the quality of the dataset to make future predictions). The S-curve has the following characteristic: in initial stage of growth it is approximately exponential; and then, as saturation begins, the growth slows, eventually remaining constant. This in our context indicates that, with age, failures grow almost exponentially and then after a certain saturation phase grow at a constant rate, eventually tapering off. **Young disks**

We note that the overall disk failure ratio (averaged over all servers) is 2.5%, very close to the aggregate disk failure ratio of 2.7% seen earlier. This is unlike the observation that Schoreder et.al [17] made, where they found steadily increasing failure rate of disk as a function of age. However, we do not have fine-grained data analogous to them in order to do a complete comparison.

Memory modules

We also looked at the reliability of memory modules (DIMM). DIMMs showed a very small failure rate (<0.03% AFR) in the first year. When we looked at all servers the CFR for DIMMs was still low (0.1%). Schroeder et. al [18] observe a much higher percentage of uncorrectable errors (1-4%) in their infrastructure. However, we have no easy way of mapping uncorrectable errors to the decision of when a memory module is replaced.

RAID controllers

RAID controllers showed a higher failure rate than memory modules. For overall RAID controllers the failure rate was 0.7% AFR and for newer RAID controllers (< 3 months) the number was close to 0.3%. Thus, similar to aggregate results shown earlier, RAID controller is a frequently failing component in servers. This is the first paper to identify RAID controller as a significant contributor to DC hardware failure.

We draw the following conclusions from these results. *First, hard disks are the number one failing component with an AFR for 2.7%. This percentage remains constant even for relatively younger disks when viewed at an yearly granularity. The next major unreliabile component is RAID controller. However, the aggregate category of failures, i.e., one that cannot be attributed to any single kind of component is dominant after hard disks.* Note that all these observations are empirical. In the next section we attempt to understand some of the reasons behind it.

3.4 Classification Trees

We attempt other techniques to explore more structures in the failure pattern. Figure 7 shows the results of a classification tree experiment on the entire population of several thousand Microsoft IT servers [6]. The goal of our statistical experiment was to explore if

[6] We did not observe any dominant predictor/factors when performing the analysis with the aggregate set of machines

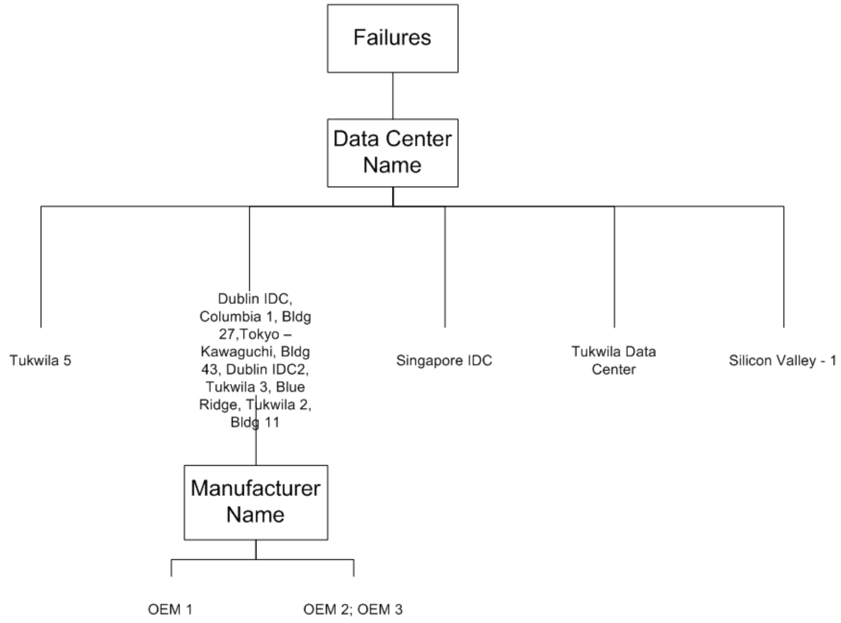

Figure 7: Classification tree for failures in MSIT servers.

failures (any type of failure indicated by a trouble ticket) could be predicted using metrics collected from the environment, operation and design of the servers in the datacenters. For this purpose we use several metrics some of which are: environment (datacenter name, manufacturer, location time zone etc.), design (number of disks, memory capacity, slots, Free virtual memory, Free physical memory etc.) and operation metrics (last OS install date, OS version, last updated time etc.). Amongst all these metrics ($>$ 50) we build a decision tree using CHAID (CHi-squared Automatic Interaction Detector) methodology [10] for adding factors to the tree (based on adjusted significance testing) in order to terminate the tree only as far as the elements added to the tree are statistically significant.

We obtain two factors namely: Datacenter name and manufacturer name. The datacenter name is an interesting result as in recent times as there has been research on studying the environment of various datacenters, the actual datacenter in which the failure is located could have an important role to play in the reliability of the system. The manufacturer is also an interesting result as different hardware vendors have different inherent reliability values associated with them (the names are intentionally anonymized). These results to the best of our knowledge are the first in the field to analyze, observe and predict failures using a wide variety of measures primarily with the goal of understanding the most dominating factors in terms of understanding failures from a statistical sense. We do not imply its use to build prediction models to replace hardware servers of a particular kind or move all datacenters to one particular location. It is purely to understand the dominating factors influencing (or not influencing) failures.

The age of the server, the configuration of the server, the location of the server within a rack [7]*,workload run on the machine, none of these were found to be a significant indicator of failures.*

4. FAILURE PATTERNS

We have so far seen various trends in hardware failures and tried to establish some patterns. In this section we examine a number of different predictors for failures. While the experiments so far were helpful in understanding high-level trends in failures it did not yield any model or insights that we can use to understand the root cause behind failures. Furthermore, the results presented thus far, while educative and informative, are not in a format that can be easily abstracted to carry out further studies by assuming failure distributions etc. In this section we perform, what we believe, is the first such predictive analysis on hardware faults in such a large scale infrastructure. While we leave a more detailed predictive reliability modeling effort for future work our aim in this section is to find key indicators of failures as well as fit the failure characteristics to well-known distributions to observe patterns.

4.1 Repaired Servers

We examine the behavior of machines once a hardware failure happens. The hypothesis is that machines that have seen at least 1 hardware incident in the past may have a different behavior and fate than the machines that do not see any hardware failure, thereby allowing us to observe an underlying structure. Identifying such properties will greatly affect our choice of actions upon seeing hardware faults, for instance, whether or not to repair them etc.

In order to understand the repair probability of machines we use the following metric, repairs per machine (RPM). We arrive at this by dividing the total number of repairs by the total number of machines. We group machines based on the number of hard disks they contain. We then look for strong indicators of failure rate in the number of server, the average age as well as number of hard disks. We plot the RPM as a function of the number of disks in a server in Figure 8. The X-axis shows the number of disks per server in each group of servers. The left X-axis shows the RPM values for each group of machines. For instance, when we look at

[7] owing to temperature/humidity gradient within rack we might have expected different failure characteristics

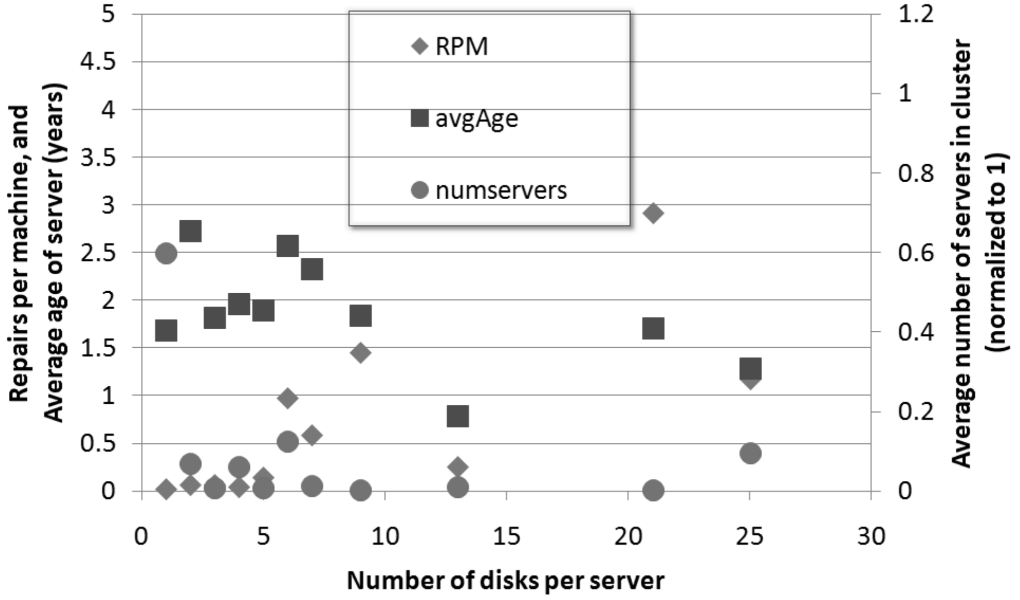

Figure 8: Repairs per machine as a function of number of disks. This includes all machines, not just those that were repaired.

all machines with 9 disks each, the average number of repairs per machine is 1.5 (diamonds off left Y-axis). Also, the average age of 9-disk machines (squares off left Y-axis) is just under 2 years. Finally, the total number of machines in the cluster (normalized to 1 by dividing by the total number of machines across all clusters) is shown with circles corresponding to the Y-axis on the right. A quick glance at the graph indicates that neither number of disks per machine, nor the age of the server is a good indicator of the number of repair events per machine. This is consistent with the decision tree experiment described in the previous section (Section 3.4).

We next divide the set of machine into two groups. Those that see at least 1 hardware incident and those who do not. We then discard the machines that do not see any hardware incident. Of the machines that see at least 1 hardware incident we again compute the RPM. Note, that by definition, each group of machines will have an RPM of at least 1 (we are only including machines that see at least 1 hardware incident). The results are shown in Figure 9. Compare this to the RPM values when all machines were put into one group in Figure 8. There is a clear emergence of some structure and pattern. First observation is that there is no trend between the age of the servers and how many repairs it sees. However, if we look at RPM values then they are clustered into two groups. Consider all clusters of machines, except for 13 and 25 disk machines. All of these can be fit into a straight line as shown in the figure, with a good fit ($R^2 > 0.9$). Thus, we can predict, for this group, the number of repairs per machine with high confidence, by just knowing the number of disks in the machine, and more importantly, irrespective of the age of the machine.

We next investigate why the other two points (corresponding to 13 and 25 disk machines) do not follow the same curve.

- **SAS vs. SATA**. One possibility is that the disks on the right are SAS and the other are SATA. Ideally, such information would be recorded along with the inventory data in the dataset, making our task easier. Unfortunately, this information is not tracked in our database. Thus, we resort to the following alternate mechanism to guess the technology of the hard disk. Basically we can tell SAS vs. SATA by the disk capacities. A SAS disk is more expensive than a SATA disk. As a result, it is used where performance and not storage is important. Thus, we would expect the average disk capacity of SAS disks to be lower than that of SATA disks. Having resolved this, we return to our original hypothesis of the 13 and 25 disks being SAS and the others being SATA. If this is true then we should be able to observe the corresponding difference in the disk capacities. However, as can been seen in Figure 10 there is no clear demarcation in the average disk capacities for the 13 and 25 disk machines (shown via triangles off the right Y-axis, normalized to 1). This rules out SAS vs. SATA as a possible explanation for the cause.

- **Improved technology.** From Figure 9 we can see that the 13 and 25 disk machines have an average age (shown by squares off the right Y-axis) lower than those of other clusters (< 2 years). It is possible that being newer disks, they have gone through a technology change resulting in higher reliability. It is also possible that in the initial period of deployment the failure rate is different than when the machine gets old [17]. If we had data beyond the 14 month period we could have observed the fate of these machines to verify this hypothesis. There might be other factors beyond our understanding i.e., datacenter, disk controller, vibrations and close packing that might result in different reliability characteristics. In ongoing work we are investigating possible causes for this.

In summary, we make the following two observations. *Firstly, there is some structure present in the failure characteristics of servers that have already seen some failure event in the past. There is no such obvious pattern in the aggregate set of machines. Second, the number of repairs on a machine shows a very strong correlation to the number of disks the machine has.* This might be intuitive given that hard-disk is the number one failing component, however, two facts make this observation interesting and worthy of further investigation. First, no such obvious relationship exists in the aggregate

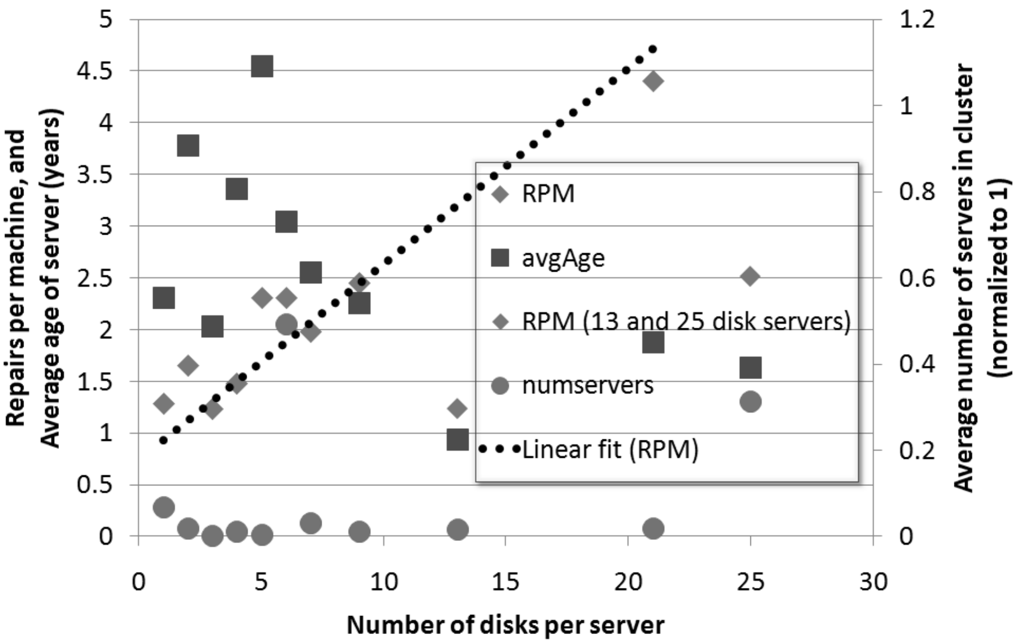

Figure 9: Repairs per machine as a function of number of disks. This is only for machines that saw at least 1 repair event.

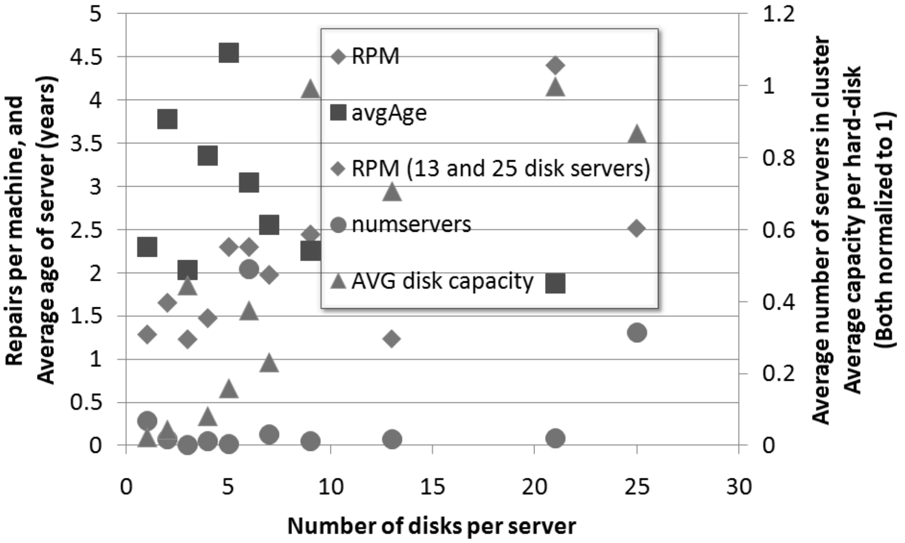

Figure 10: Repairs per machine as a function of number of disks. This is only for machines that saw at least 1 repair event. It also includes the average disk capacity of the server.

set of machines. It was only observed in machines that had already seen a failure in the past. Second, the fit is remarkably good with an R^2 of greater that 0.9.

4.2 Successive Failures

The previous section established that machines that have seen a failure in the past have some inherent structure about future failures. In this section we explore that relationship further. We begin by analysing how often do repairs happen on the same machine.

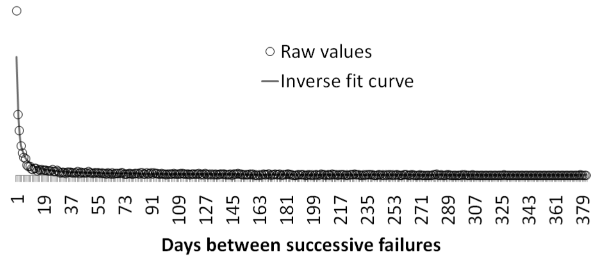

Figure 11: Distribution of days between successive failures fits the inverse curve very well.

Recall from Section 2.2 that after normalizing the server count (to 100) we found that out of 100 servers, around 9 servers see a failure in 14 months. In terms of total number of failures there were 20 failiures. We also know that around 5 machines failed only once. Thus, around 4 machines had more than 1 failure in the 14 month period. Let us examine these 4 machines and the 11 failures they see in repeat failures, more closely next. We calculate the time to next failure (not shown here) and observe that 2 of those repeat failures happen within the same day. At the other extreme, in some cases the repeat event can happen after over an year. Put another way, 20% of all repeat failures happen within a day of the first failure, and 50% of all repeat failures happen within 2 weeks of the first failure.

In Figure 11 we plot the days between successive failures and the number of times the second repair happened in the specified days between failures. The Y-axis has been omitted to maintain data privacy. However, as can ben seen qualitatively, higher Y-values towards the left of the graph suggests that a lot of successive failure events happen within a short span of the previous hardware failure on the same server. Using this large sample of failure data we analyze if there exists a statistical relationship between the days between successive failures and the number of times the second repair happened. From an exploratory standpoint we fit a set of ten standard statistical models. The goodness of fit (R^2) of these ten statistical models is shown in Table 1.

The Inverse model has the best R^2 value (represented in Figure 11 by a solid line). The general form of the inverse equation is represented by

$$D = C1 + \frac{C2}{N}$$

where D is the days between successive failures, $C1$ and $C2$ are constants, and N is the number times of second repair. The Inverse equation has a general property of diminishing returns, i.e. inverse equations observe the flat tail of the curve. The R^2 of the model indicates the efficacy of the fit of the model to describe the days between failures. *To the best of our knowledge our paper is the first to systematically and statistically study the relationship between successive failures and the number of times the second repair occurs within the time period.* An important point we would like to make is that the above results due to the inverse equation fit are indicative of the existing dataset and not about future predictions. While this would be a great starting point to model failures we feel it would be unwise to use these models to predict the days between failures as we do not yet understand in detail the reasons for such a strong relationship. We have quantified the final result (i.e. failures) which might be due to the interaction of various factors beyond our control. This result serves as further motivation for research in this field.

Table 1: R^2 values for various statistical curves fit against days between successive failures on the same machine.

Model Fit	R^2
Linear	0.178
Logarithmic	0.474
Inverse	**0.974**
Quadratic	0.292
Cubic	0.389
Compund	0.822
Power	0.771
S	0.309
Growth	0.822
Exponential	0.822

5. RELATED WORK

In this paper we analyzed the hardware reliability for a large cloud computing infrastructure. To the best of our knowledge, this is the first, research paper describing and analyzing server faults at such a large scale. There have been a number of recent efforts to understand the reliability of subsets of computing infrastructures and we acknowledge them below. Most of them however, have been around understanding the failure characteristics of individual components and not whole server reliability.

Google recently published [18] the largest study on memory modules. They found that 8% of all modules have correctable errors and the number of correctable errors per DIMM could be close to 4000 per year. They found no correlation of errors to manufacturer. They also found that temperature has a very small effect on error rates, which tend to be dominated by hard errors. The number of uncorrectable errors was 1.3% per year for few machines and upto 4% for others. Unfortunately, our current dataset contains no information on correctable or uncorrectable errors, although we do track when the module was replaced. This is typically after a fair number of errors have already been seen by the server diagnostics. In practice, we observe a DIMM replacement value of 0.1% which is significantly smaller than the number of uncorrectable errors noted by Google in their infrastructure. This leads to an interesting discussion of what denotes a fault and when should repairs ideally take place, but that is outside the scope of the current paper.

In a keynote talk [5] at the 3rd ACM SIGOPS International Workshop on Large Scale Distributed Systems and Middleware (LADIS), Jefferey Dean presented numbers and experiences from running the Google infrastructure. He observed that disk AFR is in the range 1-5% and server crash is in the range 2 to 4%. Disk AFR is in the same range as what we observe, i.e. 2.7%. We do not have access to server crashes for the machines used in this study, however, the reader may note that we observe a server failure rate of 8%. He also mentioned other single points of failure in the datacenter infrastructure including PDUs, switches etc. In this paper we only try to understand failure characteristics of servers. Building a whole datacenter reliability model consisting of all of these components is part of an ongoing effort. Google has also released a book [4] explaining how to build a datacenter. They classified all faults and found that software related errors are around 35% followed by configuration faults around 30%. Human and networking related errors are 11% each and hardware errors are less than 10%.

Schroeder et. al analyze [17] disk replacement logs from large production systems and report on the failure rate and compare that to vendor advertised values. They find huge differences form the advertised 0.88% AFR. They see upwards of 1%, 2-4% at times and upto 13% in instances. Our reported values are in the same range as quoted in their paper. We observe higher failure rate in servers that host a large number of disks (not shown here). They observe early onset of wear out in failure rates. They also did not see any huge difference in failure of SCSI, SATA, and FC drives. This result again, is similar in vein to the results we saw in Figure 10. They also observed that the time between replacement shows significant correlation, including autocorrelation and LRD (long-range-dependence). In our study we find that the failure rate of disks in the first year is very close to the failure rate for the aggregate set of machines where significant machines could be upto 3 years old. For successive repairs we observe that empirical data fits the inverse curve very well.

Another study on disk reliability was performed by Pinheiro et. al [15]. They find that disk reliability ranges from 1.7% to 8.6%. They find that temperature and utilization have low correlation to failures. However, SMART counters correlate well, for instance, scrub errors. In our current study we do not correlate SMART counters, however, we too found that environmental conditions were not a great indicator of faults. Instead, we found that datacenter location and manufacturer were the dominant indicators.

Weihand et. al look at support logs from around 40,000 commercially deployed storage systems that have around 1.8 million disks to determine the root cause behind storage system failures [11]. Their conclusion is that disk failure rate is not indicative of storage subsystem failure rate. Our current work focuses on component failure rate as well as server failure rate. In the future we are looking at incorporating this analysis into our framework. They also found that as disk capacity increases, there is no real evidence of higher failure rates. This is consistent with the results present in Figure 10. They found many bursty errors suggesting that RAID like solutions might have seen the end-of-the day and better models that do not assume independence are warranted. In ongoing work we are working on correlating RAID failures and hard disk failures co-incident on the same server to build sound models to point us in this direction.

Bairavasundram et. al [2] analyze latent errors (undetected errors) lurking inside disks that manifest upon accessing the corresponding sector. In this work we do not examine fine grained data to compare such results. Instead we rely on the detailed diagnostics to determine when it is appropriate for a hardware component to be repaired. When such a decision makes its way to a filed IT ticket, we use that to carry out our analysis.

We would like to reiterate that the aim of our current study was to discover underlying patterns in failure characteritics. Explaining the root cause behind that is part of ongoing effort. Discovering the structure, if any, in failure patterns will be an invaluable tool to help understand the nature of these events and also to assist as a modeling tool to test various datacenter designs [19, 7, 12, 1].

6. CONCLUSIONS

Demand for always available cloud computing infrastructure puts onus on the underlying software which in turn runs on commodity hardware owing to economic concerns. This make the cloud computing infrstructure vulnerable to hardware failures and the corresponding service outages. This paper is the first, to the best of our knowledge, to characterize server repair/failure rates in order to understand the hardware reliability for large cloud computing infrastructures. We find that (similar to others) hard disks are the number one replaced components, not just because it is the most dominant component but also because it is one of the least reliable. We find that 8% of all servers can expect to see at least 1 hardware incident in a given year and that this number is higher for machines with lots of hard disks. We can approximate the IT cost due to hardware repair for a mega datacenter ($>$ 100,000 servers) to be over a million dollars.

Furthermore, upon seeing a failure, the chances of seeing another failure on the same server is high. We find that the distribution of successive failure on a machine fits an inverse curve. Our initial hypothesis is that upon seeing a failure the machine makes a transition from a benign state to a new state where there is a rich structure in failure patterns. We also find that, location of the datacenter and the manufacturer are the strongest indicators of failures, as opposed to age, configuration etc.

Being a data study there a number of limitations in this analysis. For instance, we can only report based on the time period we observe. This implies that the results are potentially biased against the environmental conditions, technology, workload characteristics etc. prevelant during that period. Also, we do not investigate the cause of the fault or even the timing. We are only interested in repair events at a coarse scale and understanding what models it fits. In a fast moving technology industry we also face the perpetual danger of analysing historical logs only to find our results obsolete before we can put them to use.

In ongoing work we are looking at doing root cause analysis to explain as well as proactively predict failures. We are also working on composable models for server reliability which will help us understand the effects, for instance, if HDDs are replaced by SSDs. Finally, we are planning to incorporate whole datacenter designs, including single points of failures, for instance, PDUs, and switches into a reliability model. We hope that the results presented in this paper provides fuel to understanding and mining the behavior and causes of hardware faults in cloud computing infrastructures.

Acknowledgement

The authors wish to thank Andre Podnozov for extending technical help in mining the data during the initial part of this investigation. We would also like to thank the Microsoft operations team for giving us access to valuable piece of data enabling us to carry out this study.

7. REFERENCES

[1] M. Al-Fares, A. Loukissas, and A. Vahdat. A Scalable, Commodity, Data Center Network Architecture. In *ACM SIGCOMM*, 2008.

[2] L. Bairavasundaram, G. Goodson, S. Pasupathy, and J. Schindler. An Analysis of Latent Sector Errors in Disk Drives. In *ACM SIGMETRICS*, 2007.

[3] L. A. Barroso, J. Dean, and U. Holzle. Web Search for a Planet: The Google Cluster Architecture. In *IEEE Micro*, 2003.

[4] L. A. Barroso and U. Hölzle. The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines. In *Synthesis Lectures on Computer Architecture*, 2009.

[5] J. Dean. Large-Scale Distributed Systems at Google: Current Systems and Future Directions, 2009.

[6] D.Oppenheimer, A.Ganapathi, and D.Patterson. Why do Internet Service Fail and What Can be Done About It? In *4th USENIX Symposium on Internet Technologies and Systems*, 2003.

[7] A. Greenberg, N. Jain, S. Kandula, C. Kim, P. Lahiri, D. Maltz, P. Patel, and S. Sengupta. VL2: A Scalable and Flexible Data Center Network. In *ACM SIGCOMM*, 2009.

[8] J. Hamilton. An Architecture for Modular Data Centers. In *CIDR*, 2007.

[9] J. Hamilton. On Designing and Deploying Internet-Scale Services. In *USENIX LISA*, 2007.

[10] D. Hawkins and G. Kass. Automatic Interaction Detection. In *D.M. Hawkins (ed) Topics in Applied Multivariate Analysis. Cambridge University Press, Cambridge*, 1982.

[11] W. Jiang, C. Hu, and Y. Zhou. Are Disks the Dominant Contributor for Storage Failures? A Comprehensive Study of Storage Subsystem Failure Characteristics. In *6th USENIX Conference of File and Storage Technologies (FAST)*, 2008.

[12] R. N. Mysore, A. Pamboris, N. Farrington, N. Huang, P. Miri, S. Radhakrishnan, V. Subramanya, and A. Vahdat. PortLand: A Scalable Fault-Tolerant Layer 2 Data Center Network Fabric. In *ACM SIGCOMM*, 2009.

[13] Nicola Brace and Richard Kemp and Rosemary Snelgar. SPSS for Psychologists. In *Palgrave Macmillan*, 2003.

[14] D. Patterson, A. Brown, P. Broadwell, G. Candea, M. Chen, J. Cutler, P. Enriquez, A. Fox, E. Kiciman, M. Merzbacher, D. Oppenheimer, N. Sastry, W. Tetzlaff, J. Traupamn, and N. Treuhaft. Recovery Oriented Computing (ROC): Motivation, Definition, Techniques, and Case Studies. Technical report, UC Berkeley, March 2002.

[15] E. Pinheiro, W.-D. Weber, and L. A. Barroso. Failure Trends in a Large Disk Drive Population. In *5th USENIX Conference of File and Storage Technologies (FAST)*, 2007.

[16] How Much is the Reputation of Your SaaS Provider Worth? `http://cloudsecurity.org/2009/03/13/how-much-is-the-reputation-of-your-saas-provider-worth/`.

[17] B. Schroeder and G. A. Gibson. Disk Failures in the Real World: What Does an MTTF of 1,000,000 Hours Mean to You? In *5th USENIX Conference of File and Storage Technologies (FAST)*, 2007.

[18] B. Schroeder, E. Pinheiro, and W.-D. Weber. DRAM Errors in the Wild: A Large-Scale Field Study. In *ACM SIGMETRICS*, 2009.

[19] K. V. Vishwanath, A. Greenberg, and D. A. Reed. Modular Data Centers: How to Design Them? In *LSAP '09: Proceedings of the 1st ACM workshop on Large-Scale system and application performance*, pages 3–10, New York, NY, USA, 2009. ACM.

[20] Web Startups Crumble under Amazon S3 Outage `http://www.theregister.co.uk/2008/02/15/amazon_s3_outage_feb_2008/`.