Service Level Agreements in Web Services

Jonathan Kupferman
Department of Computer Science
University of California, Santa Barbara
Santa Barbara, CA 93106
jkupferman@cs.ucsb.edu

ABSTRACT
Service Level Agreements (SLAs) are a contract between a customer and provider which defines the performance expectations of the provider. These contracts have been prominent in many industries, but are relatively new in the area of Web Services. As companies increasingly begin to consider outsourcing infrastructure to Web Services, the importance of SLAs for those services continues to grow. There is, however, little standardization of SLAs across service providers and as a result it is difficult to compare them. The continued adoption of SLAs adds additional importance to standardization of SLAs across vendors, and should result in a standard in the near future.

Categories and Subject Descriptors
D.2.8 [**Software Engineering**]: Metrics – *performance measures*

General Terms
Management, Performance, Reliability, Legal Aspects

Keywords
Service Level Agreement, SLA, Web Services

1. INTRODUCTION
A Service Level Agreement formally defines an agreement between a provider and customer regarding the performance expected from the provider. These agreements generally layout a few key metrics under which the provider is to be judged upon along with the minimum threshold of those metrics which the provider should satisfy. The specific metrics and their definitions are dependent on the type of service being provided and the provider itself. For example, the metrics for an ISP could be network latency and packet loss whereas a metric of a call center could be the average time to answer a call. In order to ensure that providers are in fact meeting those expectations SLAs generally also define a penalty which the provider will incur if they fail to meet those metrics within a defined period of time.

There are benefits to having an SLA for both the provider and customer. One key benefit of SLAs for both parties is differentiation between competing providers. Without an SLA it can be difficult to compare two competing providers beyond the most basic means, like price. SLAs on the other hand quantify ideas like availability which give customers more information which they can use to make more informed decisions. For companies, SLAs are a balancing act between providing the strongest one possible and providing one that can be reasonably achieved. Strongest in this context translates to better service for the customer, for example higher availability or lower loss rate. Companies are motivated to provide the strongest SLA possible so that it can attract more customers and better differentiate itself from its competitors. On the other hand, if a company cannot meet its SLA it will be forced to pay out penalties to its customers. Penalties are an important incentive for providers and fail safe for customers. Providers are motivated to not violate its SLA such that it will never have to pay the penalty to its customers. Customers on the other hand are guaranteed that if the SLA is violated the provider will be forced to pay the penalty to the customers.

Companies offering SLAs have the option to offer one standard SLA to all of its potential customers, or negotiate with them on an individual basis. This choice is largely dependent on the number of partnering contracts a provider is likely to have. In the case where a company is likely to have contracts with a large number of customers, they are likely to write a single, standard, SLA for all of its customers. However, within a single SLA a company can differentiate between the different levels of service it provides. For example, companies often specify the performance it guarantees for the different levels of service it provides (e.g. gold, silver, bronze) with higher levels of service generally receiving stronger guarantees.

2. WEB SERVICE SLAS
In general, SLAs for Web Services are quite similar to those which have been used in other industries for some time now, the main difference being the metrics used. By far the most prevalent metric in Web Services is availability or uptime which is generally specified as a "number of nines" (e.g. four nines = 99.99).

2.1 History
Between 2002 and 2003 multiple papers were published which proposed formalized standards for Web Service SLAs. These standards attempt to address key issues and limitations of Web Service SLAs which were being created at the time. For example, IBM proposed "The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services"[15] The paper notes that "...SLAs are often plain natural language documents. Consequently, they must be manually provisioned and monitored, which is very expensive and slow." Another paper from members of the Computer Science Department at the University College London proposed "SLAng: A Language for Defining Service Level Agreements."[17] The proposed language "provides a format for the negotiation of QoS properties; the means to capture these properties unambiguously for inclusion in contractual agreements[.]" A common theme in both papers is that writing SLAs in plain natural language results in ambiguity and makes implementing and monitoring SLAs a difficult task. Furthermore, performing automated monitoring of SLAs would require significant work and would likely be difficult to generalize. To address these issues both propose XML-based languages which allow SLAs to be specified in more precise terms. By writing an SLA in a formalized and structured language one can then provide that SLA to automated monitoring software which can track a services compliance to the defined agreement.

2.2 Perspective

In recent years there have been an increasing opportunities for companies to begin outsourcing much of its IT labor which was traditionally done in-house. Whereas previously companies would often handle their own services like email, web hosting, and application hosting in-house, it has become increasingly feasible and cost-effective for companies to outsource these services to external vendors. One of the benefits of outsourcing this type of labor is that individual companies no longer require the assets (e.g. servers) and personnel to maintain these services; instead these are handled by the provider. Since the provider is likely performing the given service at a much larger scale it often the case that it can provide the service cheaper and more economically then if it were done in-house.

One of the biggest issues with outsourcing of any service is a potential decrease in quality of service. Although providers strive to provide the best possible service to its customers, the outsourcing of services, specifically ones that are mission critical, can be an extremely risky proposition for any company. For example, if the owners of an e-Commerce website choose to outsource web hosting to an external vendor, and that vendor experiences downtime, it is the owners of that company which incur the potential loss of revenue as a result of that downtime, not the provider.

This issue was brought to the forefront during 2008 when a few popular Web Services suffered large scale outages. In April, Amazons EC2 had a one hour outage, and then in July had an eight hour for their S3 service[3]. There were also multiple Google Apps outages[14] reported during the month of August. Thus, as in the case of the e-Commerce website, many of the sites which were being run on these services also experienced downtime. These outages brought up many questions as to the stability and reliability of the services being offered.

It is situations like the ones described above which have made Service Level Agreements a topic of interest among Web Services. There are a large contingent of services which are addressing this issue, Amazons EC2[2] and S3[5], Google Apps[8], GoGrid[7], and Mosso[10] to name a few. Given how young many of these services are, they have been actively handling the SLA issue, most have published their own within the last year. Although these services have published an SLA, there is no widely available standard for Web Service SLAs. The remainder of this paper discusses the different approaches taken by these companies in crafting their SLAs and the implications of each.

3. SURVEY

The following section is a survey of Service Level Agreements for a few large Web Services.

3.1 Google Apps SLA

On October 30th 2008 Google released an SLA[9] which covered their Google Apps suite of services (See 8.1 for full text). This suite covers some of their most popular services like GMail, Google Calendar, Google Talk, Google Docs, and Google Sites.

Google's 2008 SLA commits that the Google Apps suite of services "will be operational and available to Customer[sic] at least 99.9% of the time in any calendar month." Following the commitment, the SLA continues to more precisely define the terms used in the commitment. For example, downtime is defined as "more than a five percent user error rate" where the error rate is measured on the server side. However, downtime is only counted if it is within a period of ten consecutive minutes of downtime, or a "downtime period." To determine the uptime for any given month one must subtract the number of minutes within all downtime periods within that month from the total number of minutes in that month, and then divide the result by the total number of minutes in that month. It is noted that scheduled downtime should not be considered downtime for the purposes of calculating uptime. In the case where a month's uptime percentage is below the guaranteed 99.9% additional days are added to the customers service term (i.e. customers are provided additional days of the service for free). The number of days provided is dependent on the monthly uptime provided during that month. For example, an uptime below 99.9% but higher than 99.0% warrants three days of service, whereas uptime below 95.0% warrants 15 days of service. Customers, however, must request these additional days they are not given to them automatically.

As is often the case, the commitment itself is fairly simple; it is instead the definitions which require a greater attention to detail. For example, based on the measurement methodology a Google service could have as much as 90% downtime while still being considered "operational and available" for 100% of the time within a given month. Consider the case where a service is down for nine consecutive minutes, followed by a minute of uptime repeated for an entire month. While the service would in fact be down for 90% of the entire month, there were no periods of ten consecutive minutes where the service was down, thus based on their definitions there were no downtime periods and the service was 100% available. It is important to note that nowhere in the SLA is there a stated maximum duration of downtime (not necessarily within a downtime period) which can occur within a month. While it is unlikely for this behavior to occur, it demonstrates the importance of looking beyond the "number of nines" and more deeply into how that number is determined.

The way in which downtime is measured is perhaps the most crucial parts of the document. Measuring downtime on the server-side is the easiest way of ensuring that what is out of their control (e.g. network failure) is not considered. It has been shown that many of the failures experienced client side are not in fact a result of error on the server-side, but are instead related to the client or the network[22]. There is, however, very little information provided regarding the actual methodology used to determine the error rate or how it is computed. The way in which this rate is determined can have a large effect on whether or not a period of time is considered downtime or not. It is also reasonable assure that there can be some error associated with the monitoring software used which can skew the results of the measurements.

3.2 Amazon EC2 SLA

Following two years of beta service, Amazon announced the SLA for their Elastic Compute Cloud (EC2)[1] service on October 23rd 2008 (See 8.2 for full text). In short, EC2 allows customers to access and use some number of instances (virtualized machines) remotely without having to purchase the physical machines themselves. EC2 customers are charged only for the number of machines and the amount of time they use (per CPU per hour). Customers can change the number of machines they would like at any time and only be charged for the amount used.

In their SLA, Amazon commits to making EC2 "available with an Annual Uptime Percentage...of at least 99.95% during the

Service Year." Following their commitment, definitions are then laid out for the terms used. A "Service Year" for example is a sliding 365 day window which ends the day an SLA claim is filed. This means that customers can immediately file claims when the SLA has been broken as opposed to waiting for the end of the year. However, this also means that the customer has a maximum of one year to file a claim since any claim over 365 days is not within the current service year. The Annual Uptime Percentage statistic is calculated by "subtracting from 100% the percentage of 5 minute periods during the Service Year in which Amazon EC2 was in the state of 'Region Unavailable.'" Unavailability is when an instance (i.e. machine) has no external connectivity for a five minute period, and during that time a replacement instance cannot be launched. Thus, region unavailability is when there unavailable instances in multiple distinct locations (Availability Zones). In the case that the SLA is broken, Amazon offers customers a service credit for 10% of the bill for the most recent month in which the outages occurred. That credit however can only be applied to future bills.

In conjunction with their EC2 SLA, Amazon launched the "Service Health Dashboard" which provides their "most up-to-the-minute information on service availability" which includes their EC2 and S3 services among others. It also provides the availability information for the previous 35 days along with any incident reports that occurred during that time.

While Amazon provides a "Dashboard," they also make the monitoring criteria available publicly. By defining downtime based on a region being unavailable, Amazon allows customers to monitor the status of their service through the EC2 API, which states a region's status. This makes Amazons monitoring software much more transparent to the end user, since the user can perform the same monitoring on their end.

3.3 Mosso SLA

In 2008 Mosso, a subsidiary of Rackspace, launched a "Cloud Hosting" service which includes hosting of websites, files, and servers. Server hosting is similar to the services of Amazon EC2.

Mosso is another of the companies which offers an SLA[11] for the services which they provide.(See 8.2 for full text). Their commitment is as follows, "Mosso believes that your websites, email and databases should always be protected against unscheduled outages. Our commitment to you is that every effort will be made to keep your sites online." It continues by stating that their commitment "is not an empty promise" for two reasons, "1. The Hosting Cloud, powered by enterprise technology, is built to be highly robust[,]" and "2. We've hired great people." In the case where the above commitment is broken Mosso will give customers "1 day's hosting fee for each 60 minutes of unscheduled downtime[.]" This is counted from the time a customer reports that their website, email, or databases are offline and were not caused by a customer configuration error. They also state that there may be scheduled maintenance windows, yet they do not qualify as downtime and customers will be notified beforehand.

Mosso's SLA is in many ways significantly different then the many other Web Services SLAs. Unlike the other two SLAs, Mosso's is short, very simple, and contains no legalese, it is admittedly "super-simple." It is also important to note that at no point is an explicit availability percentage stated. This is a stark contrast to most other SLAs whose focal point is very much availability. If one however considers the stated minimum of 60 minutes of downtime per month required in order to receive a credit, then in fact Mosso's availability would be 99.86% between that of the other two SLAs.

Table 1. Maximum allowable downtime per availability

Availability %	Minutes Per Year	Minutes Per Month	Minutes Per Day
99%	5256	438	14.4
99.9%	525.6	43.8	1.44
99.99%	52.56	4.38	0.14
99.999%	5.26	0.43	0.01

4. OBSERVATIONS

4.1 Availability

To provide some idea as to how stringent an availability percentage is, Table 1 displays the maximum allowable downtime for a service during a given time period. For example, an availability of 99% allows for a maximum of 5256 minutes of downtime during the course of a year, which means that the service can be down for up to 14.4 minutes each day during that year and still have a 99% availability percentage. These values are based on the fact that each minute where the service is down is considered downtime, there is no requirement in terms of consecutive minutes required to qualify as downtime. The table demonstrates that with each additional nine, the amount of allowable downtime decreases by an order of magnitude. Once availability increases beyond "five nines" it is limited to less than a minute of downtime per year. Given such stringent requirements, such a high availability percentage is quite uncommon in Web Services.

The additional requirement of consecutive minutes required in order to be considered downtime can significantly change the number of minutes of downtime experienced. Fundamentally, downtime of some duration X results in the equivalent availability when Y failures of Z minutes occur, or when Z failures of Y minutes occur. For example, 1000 minutes of downtime results in the same availability percentage (99.81%) whether 100 failures of 10 minutes occur, or 200 failures of 5 minutes occur within a given year. The two cases however are not equivalent. Given that a failure of 10 minutes implies a failure of 5 minutes the requirement of 5 consecutive minutes is strictly stronger than that of 10 minutes. This holds for the general case, which leads to the conclusion that in the case where two SLAs have the same availability percentage and rules, one with the shorter requirement for downtime is stronger than one that has a longer requirement.

4.2 Formal Languages

A survey of the following SLAs, among many others from various Web Services, reveals that neither WSLA, SLAng or any other formal specification language is widely used today. Most Web Service SLAs continue to be written in English.

4.3 Monitoring SLAs

Given that SLAs continue to be written in English, many of the issues discussed in the WSLA and SLAng papers continue to occur within Web Service SLAs, namely monitoring. In many ways, the state if SLAs monitoring is similar to that of design by contract monitoring in software programming (e.g. JML, jContractor). There are formal languages which can be used to specify both SLAs and design by contract without the ambiguity in natural languages. Once defined in a formal language, that

specification can be used to monitor the defined contract in an automated manner. However, it is also the case that writing specifications in a formal language requires some amount of work to be invested in learning the language, and how to write a specification in it. In the case of both SLAs and design by contract, the tools are available, yet neither have been widely adopted, with companies instead choosing to address the issues themselves.

None of the three SLAs examined mentions how they monitor their software, although some form of monitoring is necessary to ensure that they are meeting the defined SLAs. All three of the SLAs require that the customer report the downtime they experience in order to receive a credit a penalty paid out by the provider. Even though customers are reporting the downtime, providers verify that the reported downtime is consistent with the information they have. In the case of Google Apps, where the measurements are made on server side, a report of downtime must coincide with an error rate of greater than 5% in order for it to be attributed to Google and not the client or network. For Amazon EC2 and Mosso it is likely that there is some form of monitoring or verification done to ensure that reported downtime by a customer is occurring and is not a result of client or network error.

5. RELATED WORK

As discussed in 2.1, there has been some discussion on formal specification languages for Web Service SLAs[15, 17]. Prior to these works, papers had been written providing analysis of SLAs in Web Services[18, 21]. These papers discuss many of the issues with SLAs at the time, including ambiguity and difficultly in monitoring. They also discuss metrics beyond availability which can be used for SLAs, including response time and concurrent requests.

More recently, [13] presents many of the benefits of Web Services, specifically for the outsourcing of business processes. [24] Presents a survey of technologies which can be used to manage Web Services, including SLAs. The work also strongly advocates the establishment of a "Web Service Management Standard" for many of the same reasons discussed here.

Both [25, 26] discuss many of the challenges of online Web Service monitoring. Each acknowledge the importance of automated monitoring and present frameworks to address the issue.

Given the regency with which many of the SLAs were created, there has been very little published work which discusses these or other Web Service SLAs.

6. CONCLUSION

As Web Services continue to become increasingly widely adopted, so does the use of SLAs for those services. Given that SLAs have benefits for both customer and provider; this is a positive trend. SLAs have also begun to foster better communication between customer and provider by encouraging providers to more publicly provide status information, as in the case of Amazons Dashboard. The largest issue that remains with SLAs is lack of standardization. While most Web Service SLAs define a value for availability, the way in which availability is defined can vary from SLA to SLA. As a result comparing availability between two SLAs is not necessarily an apples-to-apples comparison. The adoption of a formal SLA specification language could aid not only this issue, but the ambiguity and monitoring issue as well. With the continued adoption of SLAs in Web Services the ability to perform a side by side comparison of the SLAs of two similar services will become increasingly more important for both customers and providers. The standardization of SLAs is something that should occur within the very near future in order to minimize the amount of change in required for existing SLAs and allowing the new SLAs to be defined based on the standard.

7. REFERENCES

[1] Amazon Web Services. *Amazon EC2 Service Level Agreement*. Available: http://aws.amazon.com/ec2-sla/

[2] Amazon Web Services LLC. *Amazon Elastic Compute Cloud (Amazon EC2)*. Available: http://aws.amazon.com/ec2/

[3] Amazon Web Services LLC. *Amazon S3 Availability Event: July 20, 2008*. Available: http://status.aws.amazon.com/s3-20080720.html

[4] Amazon Web Services LLC. *Amazon S3 Service Level Agreement*. Available: http://aws.amazon.com/s3-sla/

[5] Amazon Web Services LLC. *Amazon Simple Storage Service (Amazon S3)*. Available: http://aws.amazon.com/s3/

[6] Amazon Web Services. *AWS Service Health Dashboard*. Available: http://status.aws.amazon.com/

[7] ServePath LLC. *GoGrid*. Available: http://gogrid.com/

[8] Google. *Google Apps*. Available: http://www.google.com/apps/

[9] Google. *Google Apps Service Level Agreement*. Available: http://www.google.com/apps/intl/en/terms/sla.html

[10] Rackspace. *Mosso*. Available: http://www.mosso.com/

[11] Mosso. *Mosso's Service Level Agreement*. Available: http://www.mosso.com/sla.jsp

[12] Engel, F. 1999. The role of service level agreements in the internet service provider industry. *Int. J. Netw. Manag.*, vol. 9, pp. 299-301,

[13] Grefen, P., *et al.* 2006. An analysis of web services support for dynamic business process outsourcing. *Information and Software Technology*, vol. 48, pp. 1115-1134,

[14] Jackson, T., "We feel your pain, and we're sorry," in *The Official Gmail Blog*. 2008. Available: http://gmailblog.blogspot.com/2008/08/we-feel-your-pain-and-were-sorry.html

[15] Keller, A. and Ludwig, H. 2003. The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services. *J. Netw. Syst. Manage.*, vol. 11, pp. 57-81,

[16] Kreger, H. 2003. Fulfilling the Web services promise. *Commun. ACM*, vol. 46, pp. 29-ff,

[17] Lamanna, D. D., *et al.* 2003. SLAng: A Language for Defining Service Level Agreements. Proceedings of the The Ninth IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS'03), 2003.

[18] Li-jie Jin, V. M., Akhil Sahai, "Analysis on Service Level Agreements," HP Laboratories, Palo Alto2002.

[19] Liu, Z., *et al.* 2001. On maximizing service-level-agreement profits. *SIGMETRICS Perform. Eval. Rev.*, vol. 29, pp. 43-44,

[20] Marilly, E., *et al.* 2002. Service level agreements: a main challenge for next generation networks. In

Universal Multiservice Networks, 2002. ECUMN 2002. 2nd European Conference on, 2002, pp. 297-304.

[21] Menasce, D. A. 2002. QoS issues in Web services. *Internet Computing, IEEE*, vol. 6, pp. 72-75,

[22] Merzbacher, M. and Patterson, D. 2002. Measuring end-user availability on the Web: practical experience. In *Dependable Systems and Networks, 2002. DSN 2002. Proceedings. International Conference on*, 2002, pp. 473-477.

[23] Papazoglou, M. P. and Georgakopoulos, D., *Service-oriented computing* vol. 46. New York, NY: ACM, 2003.

[24] Papazoglou, M. P. and van den Heuvel, W. J. 2005. Web services management: a survey. *Internet Computing, IEEE*, vol. 9, pp. 58-64,

[25] Qianxiang, W., *et al.* 2007. An Online Monitoring Approach for Web services. In *Computer Software and Applications Conference, 2007. COMPSAC 2007. 31st Annual International*, 2007, pp. 335-342.

[26] Raimondi, F., *et al.* 2008. Efficient online monitoring of web-service SLAs. Proceedings of the 16th ACM SIGSOFT International Symposium on Foundations of software engineering, Atlanta, Georgia, 2008.

[27] Sahai, A., *et al.* 2002. Automated SLA Monitoring for Web Services. Proceedings of the 13th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management: Management Technologies for E-Commerce and E-Business Applications, 2002.

[28] Services, A. W.*AWS Service Health Dashboard*. Available: http://status.aws.amazon.com/

[29] Trienekens, J. J. M., *et al.* 2004. Specification of Service Level Agreements: Problems, Principles and Practices. *Software Quality Control*, vol. 12, pp. 43-57,

[30] Windley, P. 2003. Enabling Web Services. Available: www.windley.com

8. Appendix

8.1 Google Apps SLA

Google Apps SLA. During the Term of the applicable Google Apps Agreement, the Google Apps Covered Services web interface will be operational and available to Customer at least 99.9% of the time in any calendar month (the "Google Apps SLA"). If Google does not meet the Google Apps SLA, and if Customer meets its obligations under this Google Apps SLA, Customer will be eligible to receive the Service Credits described below. This Google Apps SLA states Customer's sole and exclusive remedy for any failure by Google to provide the Service.

Definitions. The following definitions shall apply to the Google Apps SLA.

- "Downtime" means, for a domain, if there is more than a five percent user error rate. Downtime is measured based on server side error rate.
- "Downtime Period" means, for a domain, a period of ten consecutive minutes of Downtime. Intermittent Downtime for a period of less than ten minutes will not be counted towards any Downtime Periods.
- "Google Apps Covered Services" means the GMail, Google Calendar, Google Talk, Google Docs, and Google Sites components of the Service. This does not include the GMail Labs functionality or Gmail Voice and Video Chat components of the Service.
- "Monthly Uptime Percentage" means total number of minutes in a calendar month minus the number of minutes of Downtime suffered from all Downtime Periods in a calendar month, divided by the total number of minutes in a calendar month.
- "Scheduled Downtime" means those times where Google notifies Customer of periods of Downtime at least five days prior to the commencement of such Downtime. There will be no more than twelve hours of Scheduled Downtime per calendar year. Scheduled Downtime is not considered Downtime for purposes of this Google Apps SLA, and will not be counted towards any Downtime Periods.
- "Service" means the service provided by Google to Customer under the applicable Google Apps Agreement.
- "Service Credit" means the following:

Monthly Uptime Percentage	Days of Service added to the end of the Service term, at no charge to Customer
< 99.9% - ≥ 99.0%	3
< 99.0% - ≥ 95.0%	7
< 95.0%	15

Customer Must Request Service Credit. In order to receive any of the Service Credits described above, Customer must notify Google within thirty days from the time Customer becomes eligible to receive a Service Credit. Failure to comply with this requirement will forfeit Customer's right to receive a Service Credit.

Maximum Service Credit. The aggregate maximum number of Service Credits to be issued by Google to Customer for any and all Downtime Periods that occur in a single calendar month shall not exceed fifteen days of Service added to the end of Customer's term for the Service. Service Credits may not be exchanged for, or converted to, monetary amounts.

Google Apps SLA Exclusions. The Google Apps SLA does not apply to any service s that expressly exclude this Google Apps SLA (as stated in the documentation for such services) or any performance issues: (i) caused by factors outside of Google's reasonable control; or (ii) that resulted from Customer's equipment or third party equipment, or both (not within the primary control of Google).

8.2 Amazon EC2 SLA

Effective Date: October 23, 2008

This Amazon EC2 Service Level Agreement ("SLA") is a policy governing the use of the Amazon Elastic Compute Cloud ("Amazon EC2") under the terms of the Amazon Web Services Customer Agreement (the "AWS Agreement") between Amazon Web Services, LLC ("AWS", "us" or "we") and users of AWS' services ("you"). This SLA applies separately to each account using Amazon EC2. Unless otherwise provided herein, this SLA is subject to the terms of the AWS Agreement and capitalized terms will have the meaning specified in the AWS Agreement. We reserve the right to change the terms of this SLA in accordance with the AWS Agreement.

Service Commitment

AWS will use commercially reasonable efforts to make Amazon EC2 available with an Annual Uptime Percentage (defined below) of at least 99.95% during the Service Year. In the event Amazon EC2 does not meet the Annual Uptime Percentage commitment, you will be eligible to receive a Service Credit as described below.

Definitions

- "Service Year" is the preceding 365 days from the date of an SLA claim.
- "Annual Uptime Percentage" is calculated by subtracting from 100% the percentage of 5 minute periods during the Service Year in which Amazon EC2 was in the state of "Region Unavailable." If you have been using Amazon EC2 for less than 365 days, your Service Year is still the preceding 365 days but any days prior to your use of the service will be deemed to have had 100% Region Availability. Any downtime occurring prior to a successful Service Credit claim cannot be used for future claims. Annual Uptime Percentage measurements exclude downtime resulting directly or indirectly from any Amazon EC2 SLA Exclusion (defined below).
- "Region Unavailable" and "Region Unavailability" means that more than one Availability Zone in which you are running an instance, within the same Region, is "Unavailable" to you.
- "Unavailable" means that all of your running instances have no external connectivity during a five minute period and you are unable to launch replacement instances.
- The "Eligible Credit Period" is a single month, and refers to the monthly billing cycle in which the most recent Region Unavailable event included in the SLA claim occurred.
- A "Service Credit" is a dollar credit, calculated as set forth below, that we may credit back to an eligible Amazon EC2 account.

Service Commitments and Service Credits

If the Annual Uptime Percentage for a customer drops below 99.95% for the Service Year, that customer is eligible to receive a Service Credit equal to 10% of their bill for the Eligible Credit Period. To file a claim, a customer does not have to have wait 365 days from the day they started using the service or 365 days from their last successful claim. A customer can file a claim any time their Annual Uptime Percentage over the trailing 365 days drops below 99.95%.

We will apply any Service Credits only against future Amazon EC2 payments otherwise due from you; provided that, we may issue the Service Credit to the credit card that you used to pay for Amazon EC2 for the billing cycle in which the error occurred. Service Credits shall not entitle you to any refund or other payment from AWS. A Service Credit will be applicable and issued only if the credit amount for the applicable monthly billing cycle is greater than one dollar ($1 USD). Service Credits may not be transferred or applied to any other account. Unless otherwise provided in the AWSEC2 or other failure by us to provide Amazon EC2 is the receipt of a Service Credit (if eligible) in accordance with the terms of this SLA or termination of your use of Amazon EC2. Agreement, your sole and exclusive remedy for any unavailability or non-performance of Amazon

Credit Request and Payment Procedures

To receive a Service Credit, you must submit a request by sending an e-mail message to aws-sla-request @ amazon.com. To be eligible, the credit request must (i) include your account number in the subject of the e-mail message (the account number can be found at the top of the AWS Account Activity page); (ii) include, in the body of the e-mail, the dates and times of each incident of Region Unavailable that

you claim to have experienced including instance ids of the instances that were running and affected during the time of each incident; (iii) include your server request logs that document the errors and corroborate your claimed outage (any confidential or sensitive information in these logs should be removed or replaced with asterisks); and (iv) be received by us within thirty (30) business days of the last reported incident in the SLA claim. If the Annual Uptime Percentage of such request is confirmed by us and is less than 99.95% for the Service Year, then we will issue the Service Credit to you within one billing cycle following the month in which the request occurred. Your failure to provide the request and other information as required above will disqualify you from receiving a Service Credit.

Amazon EC2 SLA Exclusions

The Service Commitment does not apply to any unavailability, suspension or termination of Amazon EC2, or any other Amazon EC2 performance issues: (i) that result from Service Suspensions described in Section 7.1 of the AWSEC2; (iii) that result from any actions or inactions of you or any third party; (iv) that result from your equipment, software or other technology and/or third party equipment, software or other technology (other than third party equipment within our direct control); (v) that result from failures of individual instances not attributable to Region Unavailability; or (vi) arising from our suspension and termination of your right to use Amazon EC2 in accordance with the AWS Agreement (collectively, the "Amazon EC2 SLA Agreement; (ii) caused by factors outside of our reasonable control, including any force majeure event or Internet access or related problems beyond the demarcation point of Amazon Exclusions"). If availability is impacted by factors other than those explicitly listed in this agreement, we may issue a Service Credit considering such factors in our sole discretion.

8.3 Mosso SLA

The Mosso Service Level Promise

Our no-loophole, no-legalese SLA

Mosso believes that your websites, email and databases should always be protected against unscheduled outages. Our commitment to you is that every effort will be made to keep your sites online.

This isn't an empty promise:

1. The Hosting Cloud, powered by enterprise technology, is built to be highly robust.
2. We've hired great people.

Most importantly, we'll credit your next invoice with the equivalent of 1 day's hosting fee for each 60 minutes of unscheduled downtime (up to 100% of your Recurring Fee). It doesn't matter why—any time your websites, email or databases are offline or not functioning as a result of a failure in our systems, data center, or network is considered downtime, and we begin counting from the minute you open an incident report with our support team.

That's it—we've designed our SLA to be ultra-simple. Please note that the Mosso SLA does not cover coding or configuration errors on your part, and like all hosts, we may schedule occasional maintenance windows that will affect the availability of some services. We'll post notification of scheduled maintenance before it happens, and since we operate clusters of servers, maintenance that causes downtime should be rare.