云内参(第3期)

关于云计算可用性的定性和定量研究

(A Qualitative and Quantitative Study on Availability of Cloud Computing)

(第一部分)

陈怀临 弯曲评论创办人

北极光创投投资顾问, 云基地中云网技术顾问

Email: huailin@gmail.com

摘要:

云计算在被越来越多的个人和企业所采用, 但人们对于云计算服务在安全性, 可靠性和服务响应确定性方面的担忧也与日俱增. 虽然云服务提供商(Clouds Service Provider) 通常都会承诺SLA（Service Level Agreement）的可用性(Availability)范围等, 但许多云租户不理解可用性的内在复杂性, 因此在选择云平台时缺乏对风险进行评估的能力. 本文首次系统的定义和分析了云计算可用性的算法模型, 特别是对云计算的IaaS, PaaS和SaaS各个层次可用性的内在关系展开定性讨论. 文章的最后, 针对2008年到2012年以来AWS被外界所报道过的服务事故做了相应的统计调查和一些定量分析.

1. 云计算的挑战:

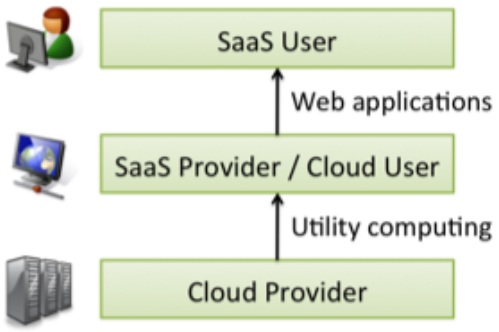
云服务在被越来越多的企业所采用. 据Gartner预测, 2013年公有云的市场份额将会以8%的增长率从2012年的1110亿美金增长至1310亿美金, 如图1所示.

图1 公有云服务市场和年增长率

在IaaS(Infrastructure as a Service)方面, 增长速度为47.3%, 市场份额为90亿美金. 2012年, IaaS增长了42.4%. 2016年, 公有云的市场大小会达到2100亿美金, 增长率为17.7%, 而在IaaS方面会保持41.3%的增长率[1].

然而, 随着大量中小型企业的CIO在考虑把公司的数据和应用迁移到云计算平台上, 伴随而来的是对云计算的服务质量(Quality of Service)的担忧.

UCBerkeley计算机系RAD实验室的Michael Armbrust等在2009年2月发表了关于对云计算服务的论文--”Above the Clouds:A Berkeley View of Cloud Computing”. 文中Berkeley提出了其理解的云计算概念模型, 并提出了云服务必须克服的10大障碍[2], 如图2所示.



	Obstacle
1	Availability of Service
2	Data Lock-In
3	Data Confidentiality and Auditability
4	Data Transfer Bottlenecks
5	Performance Unpredictability
6	Scalable Storage
7	Bugs in Large-Scale Distributed Systems
8	Scaling Quickly
9	Reputation Fate Sharing
10	Software Licensing

图2 Berkeley的云计算模型

在这10大障碍中, 1(Availability of Service), 2(Data Confidentiality and Auditability), 5(Performance Unpredictability), 6(Scalable Storage), 7(Bugs in Large-Scale Distributed Systems), 8(Scaling Quickly)都与云计算质量紧密相关. Berkeley在对可用性(Availability)的解释中, 还特别提到了DDoS攻击对云计算带来的危害和需要防范的措施.

另外, 据来自Newvem的调查数据报告, 有35%的亚马逊的AWS用户对宕机基本上没有防范措施; 40%的AWS用户没有定期做数据的备份. TeamQuest最近对许多企业的CIO做了一次调查, 接受调查的的CIO有40%的表示他们在使用云计算的时候发生了机群宕机现象[3].

2012年, 许多著名的公有云计算数据中心都发生了重大的安全事故.下面是一些典型的案例[4][5]:

*2012年2月29日和7月26日, 微软的Azure发生事故, 时间分别为长达9个小时和2.5个小时, 许多北美和欧洲的用户无法正常管理和使用其公司正常业务, 有的彻底丢失了他们最新的数据.
* 2012年6月14日, 6月29日, 10月22日和圣诞节期间的12月24日, 亚马逊AWS发生了严重云服务缓慢和崩溃无法访问的问题, 影响的租户包括许多重要的互联网公司, 例如Netflix, pInterest, twitter, Instagram等等[4]. 每次事故导致用户无法正常使用服务的时间长达9个小时和更多.
* 2012年7月10日, 著名的SaaS(Service as a Service)公司Salesforce的服务出现重大停顿. 其原因是提供Salesforce公司IaaS服务的公司(Equinix)的数据中心电源失效. Equinix据说在1分钟内就恢复了电源. 但Salesforce花费了接近9个小时来完整的恢复其相关业务.
* 2012年9月10日, 著名的DNS服务提供商GoDaddy的数据中心服务暂停. GoDaddy管理着接近5千万个域名和5百万个WEB站点. 这次服务无法正常使用长达7个小时. 其原因被解释为

路由器的数据被破坏. 也有媒体报道是GoDaddy遭遇到了强大的DDoS攻击. 但这一声称被GoDaddy否认.
* 2012年10月26日, 谷歌的App Engine云服务出现暂停, 时间长达4个小时. 事后谷歌没有发表具体原因解释.
* 2012年10月26日, 著名的云存储提供商Dropbox的服务出现暂停, 时间长达10个小时. 其具体原因不详.

由上可见, 伴随着云计算本身具备的无可争议的巨大价值, 云计算带来的诸多服务质量问题也正变得越来越明显.

因此对云计算的可用性的定性和定量分析逐渐变为一个兼有研究和工程价值的问题. 有助于帮助CIO们评估一个云计算平台.

目前学术和工业界对云计算, 特别是公有云的可用性方面还没有引起足够的重视. 缺乏这方面的定性和定量分析工作.

本文首次系统的定义和分析了云计算可用性的算法模型, 特别是对云计算的IaaS, PaaS和SaaS各个层次可用性的内在关系展开定性讨论. 文章的最后, 针对2008年到2012年以来AWS被外界所报道过的服务事故做了相应的统计调查和一些定量分析.

2. 云计算可用性(Cloud Computing Availability)

云计算可用性是一个很广义的概念. 本文定义云计算可用性如下:

云计算可用性: 包括IaaS, PaaS和SaaS各个层面服务的连接, 可靠性, 延时, 数据泄露和丢失, 网络攻击以及其他任何意外而导致租户的业务不能满足期望, 或者更严重的业务完全暂停. 云服务商通常是通过SLA(Service Level Agreement) 来量化可用性的承诺, 给出相应的Availability的数值范围, 例如,99.9%或者99.99等等.

按照云计算层次的分类[6], 我们认为云计算的Availability(简称Availability_{CS}) 包括IaaS的Availability($\text{Availability}_{IaaS}$), PaaS的Availability($\text{Availability}_{PaaS}$)和SaaS的Availability ($\text{Availability}_{SaaS}$).

$$\textbf{Availability}_{CS} = \bigcup (\textbf{Availability}_{SaaS}, \textbf{Availability}_{PaaS}, \textbf{Availability}_{IaaS})$$

我们认为, 用户最终感知的的云计算的可用性是与云计算3个层面的可用性紧密相关的.

在下面小节中, 我们首先来形式化定义一个云计算服务的可用性并做相应的算法讨论. 然后, 对云计算分层模型中IaaS, PaaS和SaaS在可用性之间的关系做理论探讨.

2.1 可用性

假定在一个采样时间范围(例如时间 T小时内)服务发生的不可用(Unavailable)次数是N. 每次不可用之前正常运行的时间定义为TBF_i(Time Before Failure). 每次用来恢复服务正常运行的时间定义为TTR_i(Time To Repair).

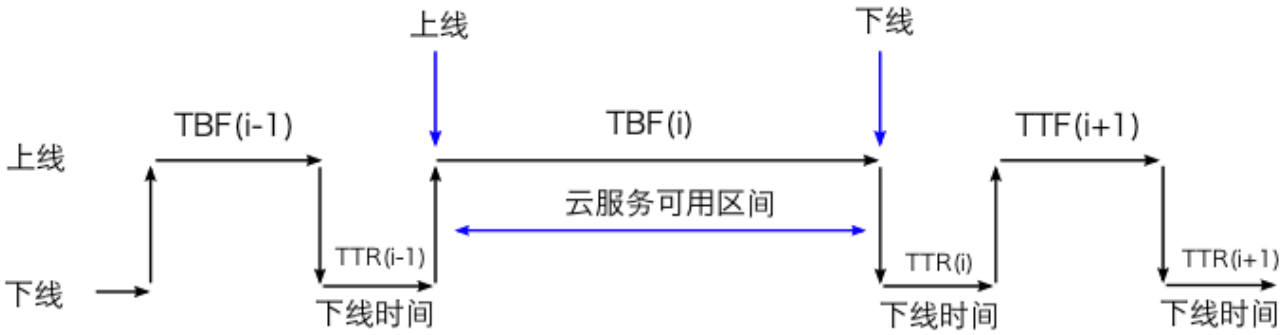


图3 云计算服务的可用性

由图3可知, 在采样时间T范围内, 服务的可用性为:

$$\text{Availability}_T = \sum_{i=1}^{N} TBF(i) \quad / \ T$$

其中, $\sum_{i=1}^{N} TBFi$ 是在时间T内, 云服务总共的可用时间.

因为 $T = \left[\sum_{i=1}^{N} TBF(i) + \sum_{i=1}^{N} TTR(i)\right]$

所以, $\text{Availability} = \sum_{i=1}^{N} TBF(i) \quad / \left[\sum_{i=1}^{N} TBF(i) + \sum_{i=1}^{N} TTR(i)\right]$

$$= \left[\sum_{i=1}^{N} TBF(i)/N\right] / \left[(\sum_{i=1}^{N} TBF(i) + \sum_{i=1}^{N} TTR(i))/N\right]$$

因此, 我们推导出在时间T小时里, 云服务的可用性为:

[公式1] $\text{Availability}_T = \text{MTBF}_T / (\text{MTBF}_T + \text{MTTR}_T)$

其中:

MTBF_T: 在时间T内, 云服务的Mean Time Before Failure[7].

MTTR_T: 在时间T内, 云服务的Mean Time To Repair[8].

根据公式1, 我们可以定义一个云服务在基于采样周期T下, 时间跨度为K下的**Mean Time Availability(MTA)**为:

[公式2] $\mathbf{MTA_K}=\sum_{i=1}^{M} Availability(i)$ **/ M, 其中M=** $\left[\frac{K}{T}\right]$

假设一个云服务的SLA取样时间T是每天, 或者说24个小时. 如果考察7224个小时的MTA, 根据上述公式, 其MTA计算方法为:

由于 $\left[\frac{K}{T}\right]$= 7224/24= 31;

$MTA_{7224小时}=[(Availability(1) + Availability(2) + ... + Availability(31)]/31$

$=\left[\sum_{i=1}^{31} Availability(i)\right]/31$

[推论1] 云服务Availability的大小与(MTTR/MTBF)的比率成反比

从 $Availability_T = MTBF_T/(MTBF_T + MTTR_T)$, 我们很容易得到,

$Availability_T = 1/(1 + MTTR_T/MTBF_T)$

定义**Mean Time Failure Ratio(MTFR)**代表(MTTR/MTBF)的比率.

$Availability_T= 1/(1+MTFR_T)$

显然, 错误失败的比率越大, 云服务的可用性就越小. 其关系曲线可以简单表示如下:

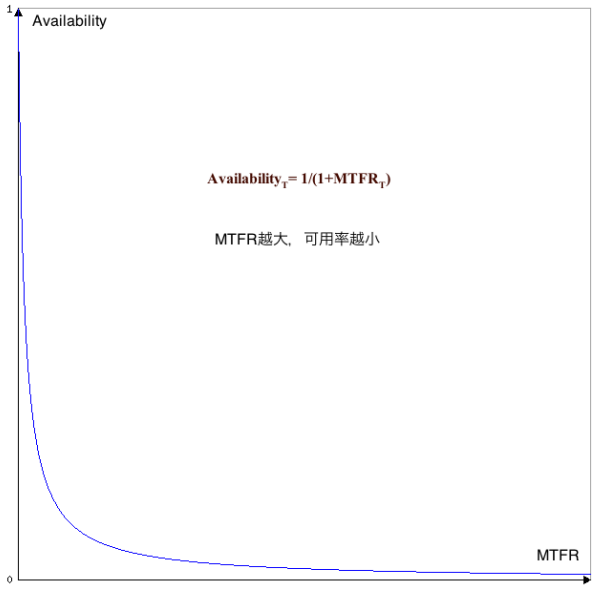


图4 MTFR与Availability的关系曲线

[推论2] 如果云服务的上线正常运行时间和下线时间是一个线性关系, 比率为a, 那么服务的可用性是一个常数$\left[\frac{a}{1+a}\right]$, 不随着采样周期T变化. 因此, SLA可以不考虑星期, 月或者年的影响.

假设 MTBF = a ×MTTR + b, 其中a, b是常量, a为云服务上线正常运行时间和下线时间的比率.

由公式1可知:

$$Availability_T = (a \times MTTR_T + b)/(a \times MTTR_T + b + MTTR_T)$$

$$=(a \times MTTR_T + b)/[(a + 1)MTTR_T + b]$$

假设b足够小, 上述推导可简化为:

$$Availability_{T=}=(a \times MTTR_T)/[(a + 1)MTTR_T]$$

$$= \left[\frac{a}{a+1}\right]$$

由此可见, 在这种情况下, 云服务的可用性是一个常量, 和采样的周期无关.

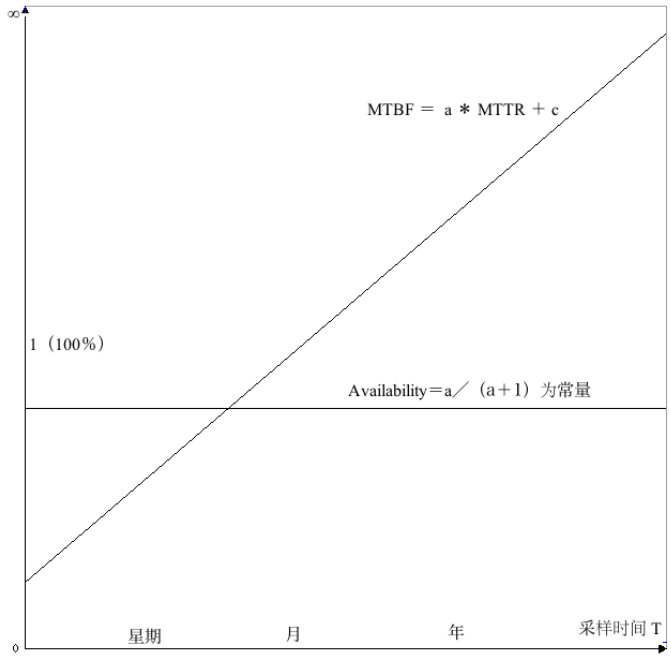


图5 MTBF和MTTR是线性关系时的Availability

[推论3] 当一个云服务运行比较稳定的时候, 云服务故障恢复的时间越短, 云服务可用性越高.

当一个上线的云服务逐渐成熟, 能稳定的运行一个固定的时间, 才出现异常. 为简单起见, 假设$MTBF_T$是一个常量σ, 意味着:

$$Availability_T = \sigma / (\sigma + MTTR_T)$$

$$= 1 / (1 + MTTR_T / \sigma)$$

可以看出, 当每次正常服务的时间是一个常量的时候, 故障发生时修复的速度越快, 时间越少, $Availability_T$ 的值就越大. 因此云服务可用性就越高.

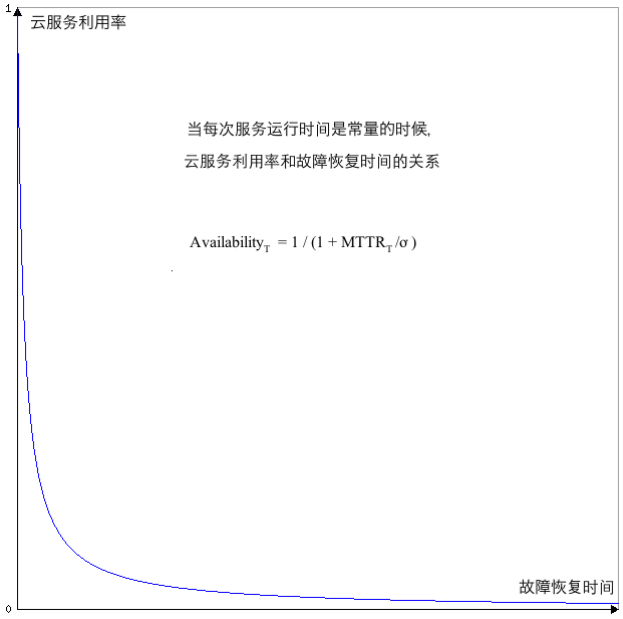


图6 MTTR与Availability的关系曲线

[案例讨论]

假设一个云服务提供商希望提供一可用性不低于某个范围值, 例如 99.9%或者99.99%等, 从而获得商业上的竞争优势.

*如果MTTR是可控的, 例如是可修护的(Repairable)部件, 具有一个修复时间上限常量Δ. 例如,云计算数据中心内的软件模块, 操作系统或者数据库的补丁, 安全漏洞等, 上述的 $Availability_T = 1/(1 + MTTR_T/MTBF_T)$ 可以简化为$Availability_T = 1/(1 + \Delta/MTBF_T)$. 那么系统的可可用性就完全依赖于MTBF, 或者说, 在时间T内服务上线的平均时间. 此时作为云计算服务提供商可以通过拉大T的取样范围(例如月或者年)和/或提高云服务的稳定性, 从而提供最大的MTBF, 以符合所期望的系统Availability参数.

* 如果MTTR是不可控的, 例如是必须更换的(Replaced)部件, 如硬盘, 服务器硬件或者电源失效等, 这些意味着MTTR的时间分布不具备一个上限常量. 这种情况下云服务提供商应该通过加大容灾处理, 1+1硬件容错等手段来确保MTTR的收敛, 并在采样时间T方面采取保守

策略, 例如(1) 对什么是不可用(UnAvailability)进行更严格的自定义, (2) 对可用性的等级采纳月, 季度或者年为单位的承诺.

[案例分析]

[例1] 一个云业务持续运行的MTBF是10,000小时, 但需要平均10个小时才能恢复正常运行,那么系统的可用性是多少?

Availability = 10,000/(10,000+10)=99.9%.

[例2] 如果要确保一个新的云业务的可用性是99.99%, 而且从内部测试可知平均运行时间大概可以保证10,000个小时才会发生错误, 那么IT运维部门必须保证在平均多长时间修复任何崩溃？

从$Availability_T = 1/(1 + MTTR_T/MTBF_T)$, 可以推导出

$MTTR_T = MTBF_T * (1 - Availability_T) / Availability_T$

因此, $MTTR_T = 10000*(1- 0.9999)/0.9999 = 1$小时.

IT部门必须在60分钟之内修复系统恢复云服务的上线, 否则就无法达到给租户承诺的SLA.

[例3] 假设一个云业务必须保证99.999%的可用性, 如果从内部测试评估认为每次业务出错恢复的时间大概为12个小时左右. 那么对业务质量控制应该是什么? 必须保证多长时间业务正常运行?

从$Availability_T = 1/(1 + MTTR_T/MTBF_T)$, 可以推导出

$MTBF_T=(MTTR_T * Availability_T)/(1 - Availability_T)$

因此, $MTBF_T= (12* 0.99999)/(1 - 0.99999)$

= 1,199,988小时 = 49,999.5天

=7,143星期=1786个月

=149年!!!!!!

这个业务必须能保证连续149年的无故障运行, 才能达到设计目标! 换言之, 5个9的设计目标是不现实的.

(To Be Continued)

[第一部分 参考文献]

1. Gartner Says Worldwide Public Cloud Services Market to Total $131 Billion

2. Above the Clouds: A Berkeley View of Cloud Computing

3. AWS Cloud Best Practice: Introduction to High Availability

4. The 10 Biggest Cloud Outages Of 2012

5. The Year in Downtime: Top 10 Outages of 2012

6.Cloud computing - Wikipedia

7. Mean time between failures

8. Mean time to repair