DOI:10.3876/j.issn.1000-1980.2011.03.021

MassCloud 云存储系统构架及可靠性机制

马玮骏[1,2],吴海佳[1],刘 鹏[1]

(1.解放军理工大学指挥自动化学院,江苏 南京 210007;2.解放军理工大学气象学院,江苏 南京 211101)

摘要:为了解决分布式存储系统的存储容量、可靠性以及效率问题,首先提出了高可靠性海量云存储系统 MassCloud 的分层体系结构,并在此构架的基础上提出了基于纠删码机制的快速编解码算法——双表法,以及基于纠删码、副本冗余和 RAID 技术相结合的 MassCloud 可靠性保证策略,并且进行了测试与分析.结果表明:双表法具有较高的编解码效率,在广域网环境下不会成为性能瓶颈;MassCloud 采用 4 组(8 块硬盘)作为每个低功耗存储节点包含的硬盘数,云存储站点层采用 3 个副本,分布式环境下冗余倍数取 2,文件分块数取 8,能够保证系统整体可靠性在 0.99 以上.

关键词:MassCloud;云存储;纠删码;副本冗余;存储可靠性

中图分类号:TP333 **文献标志码**:A **文章编号**:1000-1980(2011)03-0348-07

随着计算机与网络技术的不断发展,人们对于信息存储的要求越来越高,数据文件的大小呈指数级增长,数据存储已经发展为 TB,PB,ZB 级,存储需求对存储系统的设计和存储可靠性、存储效率形成了巨大的挑战.

纵观存储系统的发展,从早期的单磁盘存储、RAID,逐步发展为以 DAS,NAS,SAN[1] 为代表的网络存储系统,近年来比较流行的多为分布式文件系统以及存储系统.如:Antiquity[2] 是一个旨在为文件系统或者备份应用提供存储服务的大规模分布式存储系统,该系统采用安全日志技术保证数据完整性和系统的可靠性,保存日志的多个副本,同时基于拜占庭容错协议保证副本之间的一致性;Cassandra[3] 是一个用于存储结构化数据的分布式海量存储系统,该系统借鉴了很多数据库的设计和实现思想,避免了大规模分布式存储系统中的单点故障问题,旨在提供高可靠性、高性能的存储服务;文献[4]介绍了一种使用 dCache 进行存储资源管理的、在网格环境下基于 GridFTP 提供分布式存储服务的部署和实现方式;HYDRAstor[5] 是一个可扩展的商用分布式存储系统,该系统的特点在于能够自我维护,自动地进行故障恢复,对数据和网络进行重建;SandStone[6] 是一个 P2P 环境下基于 DHT 的分布式存储系统,该系统旨在解决电信基础设施与一般 DHT 之间的性能问题,为用户提供一个强一致性的、高可靠性和伸缩性的存储系统;CRAQ[7] 是一个基于对象存储技术的分布式存储系统,该系统将负载分布到多个对象副本的方式来提高系统的性能和可靠性,同时采用优化的复制链来保证副本之间的强一致性;“蓝鲸”[8-9] 是中科院计算技术研究所研究的大规模网络存储系统,该系统致力于解决高性能计算环境中的存储子系统的瓶颈问题,充分利用存储空间和并发数据传输能力,实现高性能、低成本的海量存储;文献[10]设计了一个基于 P2P 的分布式存储系统,该系统采用结构化 P2P 路由机制、动态自适应的副本管理、信任机制和激励机制为用户提供高效、可靠的分布式存储服务;分布式网络存储系统 DNSS[11] 结合了 CDN 和 P2P 的技术优势,存储节点以 P2P 方式工作以快速完成用户对文件数据的请求任务;文献[12]提出了一种基于对等计算模式的云存储文件系统通用模型,并采用 Kademlia 算法构建了原型系统 MingCloud.

从以上对存储系统以及相关可靠性策略的描述中可以看出,不同的存储系统所使用的可靠性策略存在较多的不一致性,因而对于存储系统可靠性的研究工作从未停止.本文提出一种高可靠性的云存储系统 MassCloud,对分布式环境下基于纠删码的可靠性保证机制进行了研究,根据广域网的特点提出基于纠删码的优化编解码算法——双表法,对该算法进行了分析和测试,并分析了 MassCloud 所采用的可靠性保证机制.

收稿日期:2010-06-28

基金项目:国家高技术研究发展计划(863 计划)(2008AA01A309)

作者简介:马玮骏(1980—),男,江苏南京人,讲师,博士研究生,主要从事网络管理、网格计算和分布式存储研究.E-mail:13981671@qq.com

1　MassCloud 分层可扩展构架

MassCloud 是笔者提出的一个基于云存储技术的新型分布式存储系统，该系统的设计目标在于使用最小的成本为用户提供透明、海量、高可靠的存储服务．针对设计目标，给出 MassCloud 的分层体系结构以及物理部署结构分别如图 1 和图 2 所示．该体系结构主要分为 3 层：资源节点层、云存储站点层、分布式存储层．MassCloud 采用分层的设计思想，将各类硬件、网络资源与实际的存储系统逻辑相分离，每层都处理自身所关注的主要问题，便于系统实现以及适应用户对存储系统的需求．

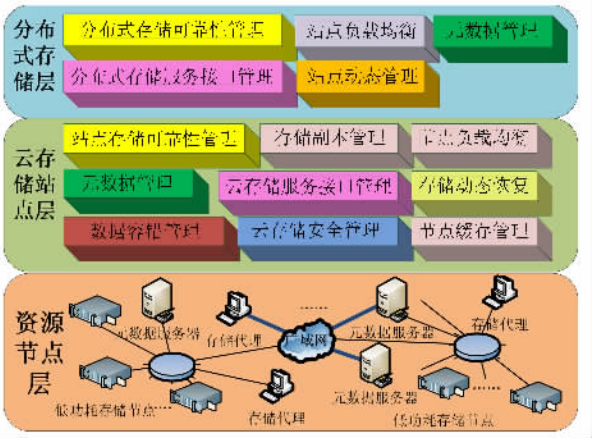

图 1　MassCloud 体系结构

Fig. 1　Architecture of MassCloud

MassCloud 的特点在于：(a)采用分层的体系结构，可以灵活地扩展整个存储体系，使得系统具有海量数据的存储能力；(b)采用低功耗存储节点作为底层存储设备，极大地降低了系统的总体硬件开销以及成本；(c)创造性地在不同层次运用不同的可靠性机制，综合运用各类可靠性机制的优势，在保证系统高可靠性的同时又兼顾了运行效率．

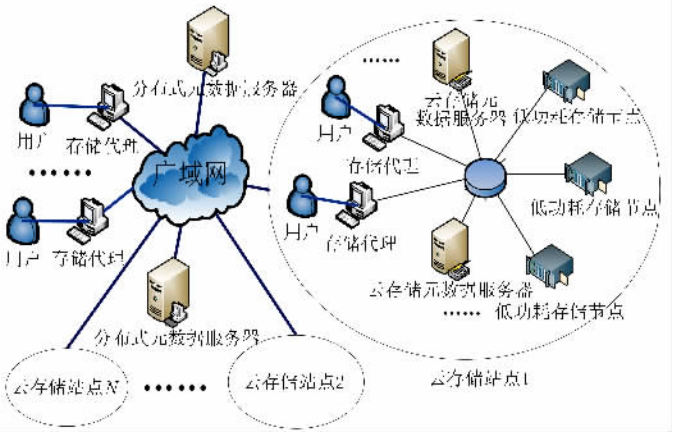

图 2　MassCloud 物理部署结构

Fig. 2　Deployment of MassCloud

1.1　资源节点层

该层主要包含各类存储资源、硬件以及网络资源，其中比较关键的是 MassCloud 特有的低功耗、低成本存储节点，该设备可以一次性挂载 4 块以上 SATA 硬盘，多个设备可以提供海量的存储空间，并且可以任意扩展．系统运行功耗在典型应用和最大应用时 5 ~ 10 W．存储节点每 TB 成本低于 1 000 元．除了低功耗、低成本存储节点，该层还包含各类元数据服务器，负责元数据的存储；存储代理，负责与用户与元数据服务器、存储节点之间的交互；资源节点层还包含大量的网络通信资源，负责存储节点、站点、元数据服务器、存储代理、用户之间通信．

1.2　云存储站点层

MassCloud 将多个低功耗、低成本存储节点相连接组成一个个存储服务器，然后再由多个存储服务器构成海量的存储空间，采用元数据服务器进行管理，通过可靠性管理、存储负载均衡、元数据管理、存储副本管理、数据容错管理、节点缓存等多种软件技术，为用户提供可靠的海量云存储服务，称之为云存储站点．云存储站点在物理上可以是一个或多个数据中心，部署在一个单位或者一个城市，为本地或者远地用户提供云存储服务．云存储站点中的元数据服务器逻辑上只有一个，但是在物理上却是多个备份相互镜像提供负载均衡，元数据服务器保存系统的元数据，负责整个云存储系统的管理．存储服务器负责具体的存储工作，数据以文件的形式存储在存储服务器上．

客户端在访问云存储站点时，首先访问元数据服务器节点，获取将要与之进行交互的存储服务器信息，

然后直接访问这些存储服务器完成数据存取.客户端与存储服务器之间直接传输数据流,同时由于文件被分成多个分块进行分布式存储,客户端可以同时访问多个存储服务器,从而使得整个系统的I/O高度并行,系统整体性能得到提高.相对于传统的分布式文件系统,云存储站点在一定规模下达到成本、可靠性和性能的最佳平衡.

1.3 分布式存储层

MassCloud认为云存储站点可能存在错误相关性,即一旦发生地震、火灾等自然灾害,某个云存储站点中的数据可能会全部毁坏,这样就必须在云存储站点层之上,再通过广域分布式存储策略构建一个分布式存储层,重点负责在整个云存储站点发生故障时,保证系统的可靠性.

分布式存储层的对象主要就是云存储站点、元数据服务器、存储代理,每个云存储站点对于分布式存储层来说就是一个海量存储设备,通过元数据服务器的管理,由存储代理为用户提供分布式抗毁的可靠存储服务.

分布式存储层采用纠删码的方式提供数据可靠性保证,因为纠删码技术在理论上被证明可靠性要优于完全副本技术[13],该层主要解决的问题便是进一步提高系统在灾难条件下的数据可靠问题.元数据服务器除了管理由纠删码技术生成的所有数据块,还提供对于所有云存储站点的负载均衡,同时可以灵活地处理云存储站点的加入和退出.考虑到减轻元数据服务器的压力,纠删码技术编码和解码的工作都由存储代理实现.

2 分布式环境下基于纠删码的可靠性保证机制及优化算法

MassCloud通过设计一种纠删能力强、编译码速度快的纠删码来满足实时传输大容量数据需要,这也是一个难点问题.根据纠错码理论[14],对(n,k,d)线性分组码(d为该分组码的最小码距),只要其中的删除位不超过$d-1$个,就可以通过适当的算法将删除位全部恢复.如果一个线性分组码能够保证从n个编码数据中任选k个都能恢复所有的删除错误,则称此码为最优纠删码.

设A为某(n,k)线性分组码C的生成矩阵,则C为最优纠删码的充要条件是A的任意k列组成的子矩阵A'均可逆[15].

在差错控制编码[16]中,最大距离可分(MDS)码的定义[17]为:设C为伽罗瓦域GF(p^r)上的(n,k)线性分组码,当且仅当其生成矩阵的任意k列线性无关时,C为MDS码.

由最优纠删码的充要条件和MDS码的定义可以推出最优线性纠删码为MDS码.MassCloud使用Reed-Solomon码(简称RS码)构造MDS码.构造MDS码的关键是寻找一个生成矩阵$A_{k\times n}$,使$A_{k\times n}$中任意k列均线性无关.RS编码技术中,比较常用的生成矩阵是范德蒙矩阵和柯西矩阵.使用范德蒙矩阵生成的编码叫范德蒙码,使用柯西矩阵生成的编码叫柯西码.文献[18]保证柯西码为最优纠删码.

柯西码的译码过程需要矩阵求逆运算,柯西码的构造中涉及求乘法逆元的运算,这些运算在实数域内存在无法整除的情况,因此需要将运算限定在伽罗瓦域内进行.伽罗瓦域对算术运算是闭包的.由此,可以设计一种快速的双表计算法.

表1 GF(2^3)元素

Table 1 Elements of GF(2^3)

指数形式	多项式形式	二进制形式
0	$\mathrm{mod}(\alpha^3+\alpha+1)=0$	000
α^0	$\mathrm{mod}(\alpha^3+\alpha+1)=\alpha^0=1$	001
α^1	$\mathrm{mod}(\alpha^3+\alpha+1)=\alpha^1$	010
α^2	$\mathrm{mod}(\alpha^3+\alpha+1)=\alpha^2$	100
α^3	$\mathrm{mod}(\alpha^3+\alpha+1)=\alpha+1$	011
α^4	$\mathrm{mod}(\alpha^3+\alpha+1)=\alpha^2+\alpha$	110
α^5	$\mathrm{mod}(\alpha^3+\alpha+1)=\alpha^2+\alpha+1$	111
α^6	$\mathrm{mod}(\alpha^3+\alpha+1)=\alpha^2+1$	101
α^7	$\mathrm{mod}(\alpha^3+\alpha+1)=\alpha^0=1$	001
α^8	$\mathrm{mod}(\alpha^3+\alpha+1)=\alpha^1$	010
□	□	□

从表1可以看出,GF(2^m)中的元素可以表示为指数形式,或者二进制形式.进行伽罗瓦域内的算术运算时,指数形式适合于乘除法运算,二进制形式适合于加减法运算,因此,可以将伽罗瓦域内的算术运算转变为查表计算.

首先构造两张表T_1和T_2.表T_1存放由多项式形式到二进制形式的对应关系,T_1的第i个元素T_{1i}表示α^i对应的二进制数值;表T_2存放由二进制形式到多项式形式的对应关系,若$T_{1j}=i$,则$T_{2i}=j$,即表T_2中的第i个元素T_{2i}表示二进制形式伽罗瓦域内的元素i对应的多项式形式的指数,对表T_1循环搜索一次就可构造出表T_2.

若将1Byte的数值看作GF(2^8)内的元素,则数值间的加减法运算直接转变为进行异或运算,数值间的乘除法运算利用表T_1和T_2转变为

$$xy = T_1((T_{2x}+T_{2y}) \bmod (2^m-1)) \qquad x/y = T_1((T_{2x}-T_{2y}) \bmod (2^m-1))$$

可见，使用双表法完成 GF(2^m)内元素的一次乘(除)运算只需要 3 次查表运算、1 次加(减)运算和 1 次模运算，合计 5 次基本操作，因而计算效率非常高.

将双表法应用于分布式环境下的分布式存储层，网络数据传输存在一定的时耗. 由于文件分块合并算法与数据的上传/下载可并行进行，因此只有双表法的时耗超过网络传输时耗，才会对系统性能产生影响. 通过实验方法统计对不同长度的文件进行分块与合并所需要的时耗，并计算相应的编解码速率. 实验计算机配置：CPU 为 Intel 双核 2.33 GHz，内存为 2 GB. 实验数据见表 2.

表 2　时耗对比

Table 2　Comparison of time consumption

文件长度/Mb	$k=4, n=6$				$k=5, n=10$				$k=8, n=12$			
	分块		合并		分块		合并		分块		合并	
	时耗/s	编码速率/(Mb·s^{-1})	时耗/s	译码速率/(Mb·s^{-1})	时耗/s	编码速率/(Mb·s^{-1})	时耗/s	译码速率/(Mb·s^{-1})	时耗/s	编码速率/(Mb·s^{-1})	时耗/s	译码速率/(Mb·s^{-1})
1	0.453	17.660	0.539	14.842	0.711	11.251	0.537	14.897	0.547	14.625	0.687	11.644
10	4.383	18.252	5.128	15.600	6.772	11.813	5.347	14.961	5.219	15.328	7.001	11.426
100	43.534	18.376	49.677	16.104	68.198	11.730	54.258	14.744	51.318	15.589	68.258	11.720

从表 2 可以看出：当 $k=4, n=6$ 时编码平均速率约为 18 Mb/s，译码平均速率约为 15 Mb/s；当 $k=5, n=10$ 时，编码平均速率约为 11 Mb/s，译码平均速率约为 14 Mb/s；当 $k=8, n=12$ 时，编码平均速率约为 15 Mb/s，译码平均速率约为 11 Mb/s. 对于分布式环境，多为广域网，双表法的编/译码速率远远高于接入网速率，因此，算法对系统性能不会产生影响. 编码速率随 k 的增大而增大，随 n 的增大而减小；译码速率随 k 的增大而减小，而与 n 的取值无关.

3　基于纠删码、副本冗余和 RAID 相结合的可靠性机制分析

MassCloud 在分布式存储层使用纠删码，在云存储站点层使用副本冗余，在资源节点层使用 RAID 分别保证各个层次的可靠性，下面就各个层次的可靠性以及系统可靠性进行分析.

3.1　RAID 可靠性分析

RAID 机制迄今为止已经发展了较长时间，技术相对较为成熟，常用的 RAID 级别主要有 RAID0，RAID1，RAID0 + 1，RAID10，RAID3，RAID5，RAID6，RAID7 等[19-20]. 综合考虑成本、可靠性、访问效率、恢复时间、实现复杂度等因素，RAID0，RAID1，RAID0 + 1，RAID3，RAID5 的可靠性相对不高，RAID6 和 RAID7 的实现成本较高，虽然 RAID5 + Hotspare 可提供较高的可靠性，但实现机制较复杂. RAID10 能够提供较高的可靠性，并且效率较高，虽然磁盘利用率较 RAID5 + Hotspare 低，但是考虑低功耗存储节点本身的成本较低，故可以接受，因此 MassCloud 的资源节点层采用 RAID10 作为低功耗存储节点的可靠性保证技术.

假设构成一个低功耗存储节点的 RAID10 的磁盘组数为 p(2 个磁盘一组)，第 j 组磁盘中每个磁盘的可靠性为 $R_{\mathrm{HDD}j}$，构成一个低功耗存储节点的 RAID10 可靠性为 R_{RAID10}，则 RAID10 的可靠性模型为

$$R_{\mathrm{RAID10}} = \prod_{j=1}^{p} [1 - (1 - R_{\mathrm{HDD}j})^2] \qquad (1)$$

假设每组磁盘的可靠性相同，取 $R_{\mathrm{HDD}j} = 0.9$，$p = 1 \sim 10$，则构成一个低功耗存储节点的 RAID10 的可靠性如图 3 所示.

从图 3 可以看出，采用 RAID10 方式构建的低功耗存储节点的可靠性随磁盘组数的增加而降低，因此每个低功耗存储节点包含的磁盘组数不宜太多，当取 4 组(8 块硬盘)时，低功耗节点的可靠性约为 0.96，在提供足够的存储容量与可靠性之间进行权衡，MassCloud 采用 4 组(8 块硬盘)作为每个低功耗存储节点包含的磁盘数.

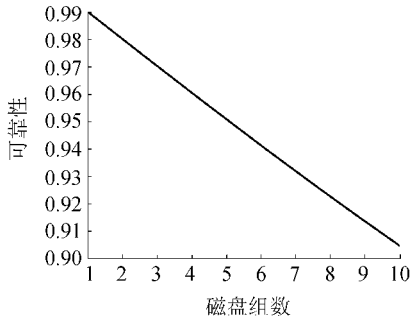

图 3　采用 RAID10 的低功耗存储节点可靠性

Fig. 3　Reliability of low cost storage node using RAID10

3.2　副本冗余可靠性分析

副本冗余技术即保存多个要存储的数据的完整副本，一个云

存储站点由多个低功耗存储节点构成,设云存储站点可靠性为 R_{COPY},低功耗存储节点可靠性为 R_{RAID10},数据冗余倍数为 r,对于云存储站点使用副本模式的冗余存储,在不同的低功耗存储节点上保存数据的多个完整副本,云存储站点中任何低功耗存储节点有效即可满足云存储站点有效,因此,由低功耗存储节点构成的云存储站点的可靠性模型为

$$R_{\mathrm{COPY}} = 1 - (1 - R_{\mathrm{RAID10}})^r \tag{2}$$

取 $R_{\mathrm{RAID10}} = 0.9$,$r = 1 \sim 5$,则采用副本冗余方式的云存储站点可靠性如图4所示.

从图4可以看出,当副本数等于1和2时,云存储站点的可靠性小于0.98;当副本数大于或等于3时,云存储站点的可靠性大于0.99,而且可靠性随副本数增加已变化不明显,考虑云存储站点的资源利用率,兼顾可靠性,MassCloud 在云存储站点层采用3个副本.

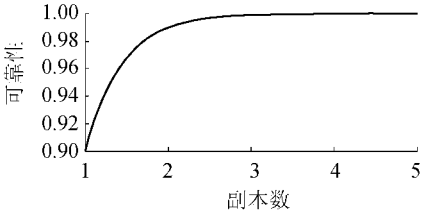

图4 采用副本冗余方式的云存储站点可靠性

Fig. 4 Reliability of cloud storage site using duplicate redundancy

3.3 纠删码可靠性分析

将 k 个文件分块数据生成为 $n(n > k)$ 个带有一定冗余的编码数据,然后将 n 个冗余的编码数据分散存储到多个云存储站点中,其中任意 k 个部分可以用来恢复原始数据.设分布式环境下使用纠删码(MassCloud 采用基于双表计算的快速编码技术)的系统可靠性为 R_{RS},云存储站点可靠性为 R_{COPY},纠删码数据冗余倍数为 $s(s = n/k)$,文件分块数为 k,只要分布式环境下任意 $m(m \geqslant k)$ 个云存储站点有效,即可满足分布式存储层的数据有效.因此,分布式存储层的可靠性模型为

$$R_{\mathrm{RS}} = \sum_{i=0}^{ks-k} C_{ks}^{i} R_{\mathrm{COPY}}^{ks-i} (1 - R_{\mathrm{COPY}})^i \tag{3}$$

式中 C 为组合运算符.取 $R_{\mathrm{COPY}} = 0.9$,$k = 4 \sim 10$,$s = 1 \sim 5$,则分布式环境下采用纠删码方式的分布式存储层可靠性如图5所示.

从图5可以看出,在低冗余度条件下(冗余倍数小于2),纠删码方式提供较低的系统可靠性,并且文件分块数越多,可靠性越低;当冗余度大于2时,纠删码将提供较高的可靠性,并且变化不再明显,保持在0.999以上.分布式存储层的冗余度以及文件分块数将在下文给出.

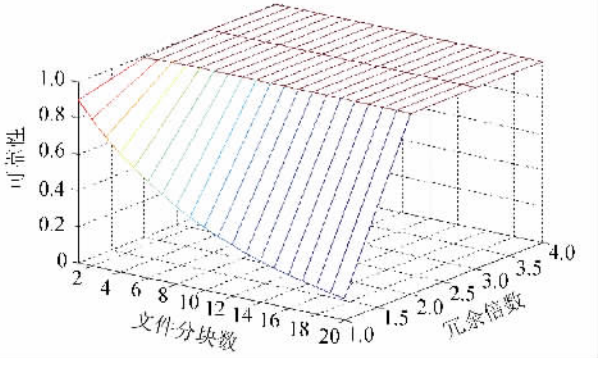

图5 分布式环境下采用纠删码方式的分布式存储层可靠性

Fig. 5 Reliability of distributed storage using erasure code

3.4 系统可靠性分析

根据以上对 RAID、副本冗余和纠删码等技术的分析,可以得出 MassCloud 可以提供的系统可靠性模型.设 MassCloud 系统可靠性为 $R_{\mathrm{MassCloud}}$,将式(1)和式(2)代入式(3)得

$$R_{\mathrm{MassCloud}} = \sum_{i=0}^{ks-k} C_{ks}^{i} \left\{1 - \left\{1 - \prod_{j=1}^{p} [1 - (1 - R_{\mathrm{HDD}j})^2]\right\}^r\right\}^{ks-i} \left\{1 - \prod_{j=1}^{p} [1 - (1 - R_{\mathrm{HDD}j})^2]\right\}^{ri} \tag{4}$$

取 $R_{\mathrm{HDD}j} = 0.9$,$p = 4$,$k = 1 \sim 20$,$s = 1 \sim 4$,$r = 3$,假定云存储站点可靠性为定值,考察系统可靠性与分布式环境下冗余倍数以及文件分块数的关系如图6所示.

假设在云存储站点副本数取3,可以看出分布式环境下 MassCloud 的可靠性与冗余倍数以及文件分块数之间的关系,当冗余倍数等于1时(即不冗余),MassCloud 可靠性随文件分块数增加而降低,当冗余倍数大于或等于2时,系统可靠性随文件分块数增加而增加,当文件分块数达到5块以上时,MassCloud 可靠性大于0.999,并且增加不再明显,因此综合考虑存储空间资源利用率以及系统可靠性,MassCloud 冗余倍数取2,文件分块数取8.

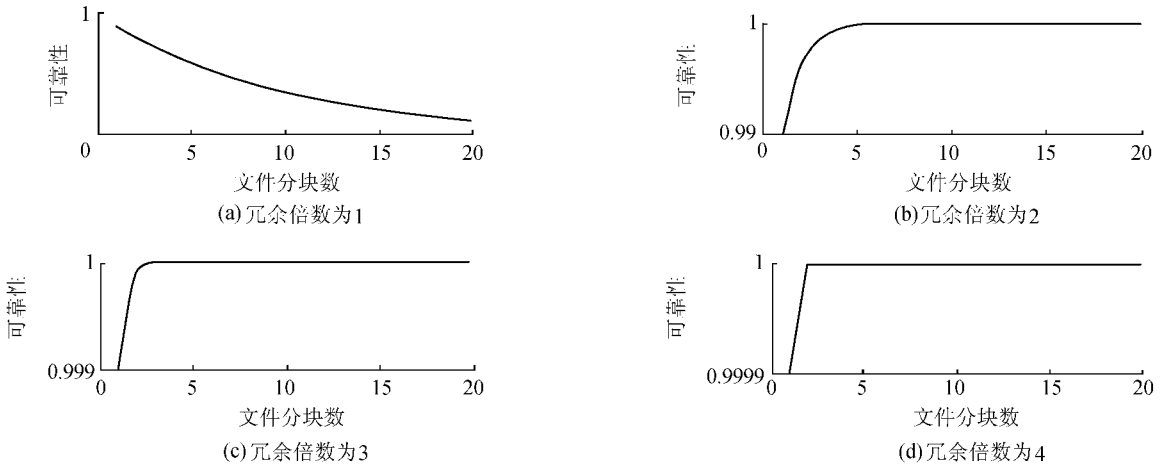

图 6　系统可靠性与分布式环境下冗余倍数以及文件分块数的关系

Fig. 6　Relationship among system reliability, times of duplicate redundancy in distributed environment and number of file segments

4　结　　语

本文介绍了高可靠性云存储系统 MassCloud 的构架以及其可靠性机制，提出了基于 RS 纠删码的快速编解码算法——双表法，该方法在分布式环境中的广域网体现出较好的性能，并且不会成为系统的效率瓶颈. 通过对各个 MassCloud 的各个层次所采用的可靠性机制的分析，在保证系统高可靠性的同时兼顾系统存储空间的利用率，得出了保证各个层次可靠性的基本参数，使得低功耗存储节点的可靠性可以达到 0.96，云存储站点和分布式存储可靠性都可以达到 0.99 以上.

参考文献：

[1] 张江陵，冯丹. 海量信息存储[M]. 北京：科学出版社，2000：95-107.

[2] WEATHERSPOON H，EATON P，KUBIATOWICZ J，et al. Antiquity：exploiting a secure log for wide-area distributed storage[C]//Proceedings of the 2nd ACM SIGOPS/EuroSys European Conference on Computer Systems 2007，Lisbon，Portugal. New York：ACM，2007：371-384.

[3] LAKSHMAN A，MALIK P. Cassandra：structured storage system on a P2P network[C]//Proceedings of the 28th ACM Symposium on Principles of Distributed Computing. New York：ACM，2009：5-5.

[4] BEHRMANN G，FUHRMANN P，GRONAGER M，et al. A distributed storage system with dCache[J]. Journal of Physics，2008，119(6)：2014-2025.

[5] DUBNICKI C，GRYZ L，HELDT L，et al. HYDRAstor：a scalable secondary storage[C]//Proccedings of the 7th Conference on File and Storage Technologies. California，USA：USENIX Association Berkeley，2009：197-210.

[6] SHI Guang-yu，CHEN Jian，GONG Hao，et al. SandStone：a DHT based carrier grade distributed storage system[C]//2009 International Conference on Parallel Processing. Vienna，Austria：IEEE Computer Society，2009：420-428.

[7] JEFF T，MICHAEL J. Freedman bject storage on CRAQ：high-throughput chain replication for read-mostly workloads[C]//Proceedings of the 2009 Conference on USENIX Annual Technical Conference. California，USA：USENIX Association Berkeley，2009：11.

[8] 杨德志，许鲁，张建刚. 蓝鲸分布式文件系统元数据服务[J]. 计算机工程，2008，34(7)：4-6.((YANG De-zhi，XU Lu，ZHANG Jian-gang. Blue whale distributed file system metadata service[J]. Computer Engineering，2008，34(7)：4-6.(in Chinese))

[9] 张敬亮，张军伟，张建刚，等. 蓝鲸文件系统中元数据与数据隔离技术[J]. 计算机工程，2010，36(2)：28-30.(ZHANG Jing-liang，ZHANG Jun-wei，ZHANG Jian-gang，et al. IsolatI/On technology of metadata from data in blue whale file system[J]. Computer Engineering，2010，36(2)：28-30.(in Chinese))

[10] 田荣华，卢显良，侯孟书，等. P2P 分布式存储系统[J]. 计算机科学，2007，34(6)：47-48.(TIAN Rong-hua，LU Xian-liang，HOU Meng-shu，et al. P2P-based distributed storage system[J]. Theoretical Computer Science，2007，34(6)：47-48.(in Chinese))

[11] 马军，郑烩，殷保群. 基于 CDN 和 P2P 的分布式网络存储系统[J]. 计算机应用与软件，2010，27(2)：50-52.(MA Jun，YAN Bao-qun，ZHENG Quan. CDN and P2P-BASED distributed network storage system[J]. Computer Applications and Software，2010，27

(2):50-52.(in Chinese))
[12] 吴吉义,章剑林,傅建庆,等.基于 Kademlia 的云存储系统数据冗余方案研究[J].电信科学,2011,27(2):68-73.(WU Ji-yi, ZHANG Jian-lin, FU Jian-qing, et al. Study on data redundancy scheme in kademlia cloud storage system [J]. Telecommunications Science, 2011, 27(2): 68-73.(in Chinese))
[13] WEATHERSPOON H, KUBIATOWICZ J. Erasure coding Vs. replication: a quantitative comparison[C]//IPTPS 2002. London: Springer-Verlag, 2002: 328-338.
[14] HAMMING R W. Error detecting and error correcting codes[J]. The Bell System Technical Journal, 1950, 29(2): 147-160.
[15] RIZZO L, VICISANO L. A reliable multicast data distribution protocol based on software FEC techinques[C]//Proceedings of the Fourth IEEE HPCS'97 Workshop. New York: ACM, 1997: 116-125.
[16] 林舒,科斯特洛.差错控制编码[M].晏坚,译.北京:机械工业出版社.2007:9-33.
[17] WILLIAMS F J, SLONE N J. Theory of error-correcting codes[M]. Amsterdam, The Netherlands: North-Holland, 1977.
[18] BLOMER J, MITZENMACHER M, SHOKROLLAHI A. An XOR-based erasure-resilient coding scheme(Report, NO. TR-9504)[R]. Berkeley, California: The International Computer Science Institute, 1995.
[19] TOM C. 存储区域网络设计[M].邓劲生,李宝峰,李蕾,等,译.北京:电子工业出版社,2005.
[20] ZHOU Ke, HUANG Yong-feng. Disk tree-a case of parallel storage architecture to improve performance in random access patten[J]. Chinese Journal of Electronics, 2005, 14(1): 39-44.

Architecture and reliability of cloud storage system MassCloud

MA Wei-jun[1,2], WU Hai-jia[1], LIU Peng[1]

(1. *Institute of Command Automation, PLA University of Science and Technology, Nanjing* 210007, *China*;
2. *Institute of Meteorology, PLA University of Science and Technology, Nanjing* 211101, *China*)

Abstract: The hierarchic architecture of the cloud storage system MassCloud with high reliability and mass data storage capability was proposed to solve the current problems of distributed storage system, that is, storage capacity, reliability and efficiency. On such a basis, a double-table method was designed based on the erasure code to gain high encoding/decoding speed in order to suit the distributed storage environment. The reliability of MassCloud was guaranteed by a policy combining the erasure code, duplicate redundancy and RAID. The relevant tests were performed and analyzed. The results show that the double-table method has high encoding/coding efficiency and may not become a bottleneck under the WAN environment. The whole reliability of MassCloud can be ensured to be above 0.99 by using 4 groups of disks (8 pieces of disks) with low cost storage level, 3 replicas for cloud storage level, 2 times redundancy in distributed environment and 8 pieces of file division.

Key words: MassCloud; cloud storage; erasure code; duplicate redundancy; storage reliability