第32卷 第18期 计 算 机 工 程 2006年9月
Vol.32 № 18 Computer Engineering September 2006
·软件技术与数据库· 文章编号：1000—3428(2006)18—0072—03 文献标识码：A 中图分类号：TP311

蓝鲸分布式文件系统网络容错的软件技术

范 勇，张建刚，许 鲁
(中国科学院计算技术研究所，北京 100080)

摘 要：蓝鲸分布式文件系统(BWFS)克服了传统存储模式在性能、容量、共享、可扩展性、可管理性等方面的局限性，通过采用连接复制、通道切换、请求重构等网络容错软件技术，在无须额外硬件支持的前提下提高可用性。文章针对系统可用性，分析了BWFS所面临的网络相关故障，从系统软件角度阐述了BWFS所采用的网络容错技术，并对其效能进行了相应的测试比较。

关键词：BWFS；网络容错；连接复制；通道切换；请求重构

Soft-technology of Network Fault-tolerant for Blue Whale Distributed File System

FAN Yong, ZHANG Jiangang, XU Lu
(Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080)

【Abstract】Blue whale distributed file system(BWFS) overcomes the shortcomings of traditional storage system in performance, capability, shareability, scalability, manageability and so no. It enhances the availability by adopting the soft-technology of connection-copy, channel-switch, and request-rebuild for network fault-tolerant without extra hardware support. Aiming at the system availability, the article analyses the fault about network faced by BWFS, expatiates the technology of network fault-tolerant adopted by BWFS from the view of system software, tests and compares its efficiency .

【Key words】BWFS; Network fault-tolerant; Connection-copy; Channel-switch; Request-rebuild

计算机系统在经历了从计算瓶颈向内存瓶颈的转变之后，正逐渐向存储瓶颈的方向发展，越来越多的应用对存储系统提出越来越高的要求。蓝鲸分布式文件系统(Blue whale distributed file system，BWFS)是国家高性能计算机工程技术研究中心自主设计开发的用于解决系统存储瓶颈的海量网络存储系统的文件系统部分。它基于普遍使用的IP协议，采用SAN与NAS融合的架构，分离元数据与数据的传输通道(元数据经由元数据服务节点传输处理，数据读写直接访问存储节点)。与传统存储模式相比，除了在性能、容量、共享、可扩展性、可管理性等方面的优势外，BWFS采用连接复制、通道切换、请求重构等网络容错软件技术，在无须额外硬件支持的前提下提高系统可用性。

本文分析了BWFS所面临的网络相关故障，从系统软件角度阐述BWFS所采用的容错技术，并对其效能进行了相应的测试比较。

1 网络容错概览

1.1 网络故障分类

目前关于网络故障的划分标准很多，依据网络故障的原因划分为：

(1)网络硬(固)件故障，如网卡故障、网线故障、交换机故障等；

(2)网络软件故障，如网卡驱动异常、协议栈异常、其他软件错误关联异常等。

依据网络故障对系统的影响划分为：

(1)性能骤降，表现为某项(些)网络性能急剧下降，虽然可以收发数据包，但与正常性能指标相去甚远；

(2)连接中断，表现为某个(些)曾经建立的逻辑网络连接(如TCP连接)异常中断，但节点之间可通过其他方式通信；

(3)网络分割，表现为系统中部分节点之间的原有通信受阻(通常是物理中断)，不能收或发相应数据包。

上述划分并非绝对，不同的划分标准导致不同的划分结果，各种网络容错技术往往针对特定的网络故障，它们可能是某些其他标准(或交叉标准)的划分结果。

1.2 热备容错

所谓热备是指当系统中的某一部件发生故障时，使用该部件的替代品立即接替它继续工作。热备隐含着冗余的思想，但冗余本身并不能容错，热备需要软件或固件的支持，用于系统故障监测、工作切换及故障恢复等。常用的热备模式有如下几种(A、B实现相同的系统功能)：

(1)主从热备，系统正常运行时只有A参与系统工作，当且仅当A发生故障时B才接替A参与系统工作；

(2)镜像热备，系统正常运行时A与B各自完成相同的工作，当一个部件出现故障时，另一部件的工作不受影响；

(3)互补热备，系统正常运行时A与B分担系统工作，当A发生故障时，B接管A的工作，反之亦是如此。

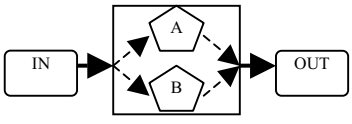

图1 热备示意图

基金项目：国家“863”计划基金资助项目(2002AA112010)
作者简介：范 勇(1979—)，男，硕士生，主研方向：大型网络存储系统；张建刚，副研究员；许 鲁，研究员
收稿日期：2005-11-24 **E-mail：**fanyong@ict.ac.cn

热备是计算机领域中十分常见的容错技术，对于网络容错而言，多网卡的容错绑定是较为普遍的网络容错手段。

1.3 协议容错

网络协议定义了数据在节点之间的传输管理规范，针对可能出现的传输异常，如误码、乱序、丢包等，部分协议采取相应的容错措施，主流的 TCP/IP 协议便是如此。

校验和是一种基本的协议容错手段，用于传输误码的检验和校正。此外，IP 协议支持对乱序传输碎片包的排序重组功能，TCP 协议提供用于处理传输超时、网络丢包等情况的重传机制。

协议容错作为多数系统的默认支持提供了基本的网络容错功能，对于提供更高可用性的分布式系统而言需要额外的网络容错支持。目前大多数分布式系统的网络容错是通过复杂的硬件冗余热备实现的，如 Caltech 的 RAIN System[3]。这类容错方式基于冗余度较高的网络拓扑模型，在很大程度上解决了硬件原因引起的网络故障,对处理网络分割尤为有效。其所带来的问题是较高的硬件成本及工程实施上的困难，且对硬件系统运行良好的网络异常(如网络性能骤降异常、逻辑连接中断异常等)处理欠佳,而这正是 BWFS 网络容错软件技术所要解决的问题。

2 BWFS 网络容错的软件技术

如前所述，网络分割的影响虽重，但就 BWFS 而言，其发生的概率很小，BWFS 中影响较大的两种网络异常如下:

(1)通道读性能骤降化异常，表现为在网络重负载情况下针对同一TCP连接通道写流量累计达到一定程度后(百 GB 量级),该通道上的网络读性能急剧下降到正常读性能的 20%以下。实测表明此种网络异常在标准 NBD(Network Block Device)等系统中同样存在

(2)通道连接软故障中断异常，表现为客户节点与存储节点之间的 TCP 连接在网络硬件运行良好的情况下异常中断(可能是系统软件故障，也可能是误操作所为，系统底层无法区分)，导致基于该通道的数据传输失败。同样的问题存在于标准 Nbd 等其他 TCP 相关系统中

针对上述异常，BWFS 采用连接复制、通道切换、请求重构等软件技术加以容错。

2.1 连接复制

在 BWFS 中,客户节点与存储节点之间的数据传输通过 TCP 连接通道进行，上述两种对 BWFS 影响较大的网络异常最终都表现为该通道异常，加强该通道的可靠性至关重要。受热备思想启发，可以对该通道做软件“热备”，以便当异常发生时可以使用备用通道继续工作。

通道本身占用系统资源，在没有上述网络异常时，备用通道对系统资源占用是无益的，而当上述网络异常发生时，之前建立的备用通道可能亦受网络异常影响而不再可靠；另一方面在多存储节点的 BWFS 中，客户节点与不同的存储节点之间存在不同的连接通道，不区分情况地为每一个连接通道都建立一个备用通道是对系统资源的浪费。因此这种通道“热备”是动态按需分配的，即监测到系统发生上述网络异常时才针对故障通道通过复制方式建立备用通道。

所谓复制是因为对于每一个传输通道而言，其在建立时初始化了一些参数，包括存储节点 IP 地址、目标端口、认证以及其他一些 BWFS 信息，这些参数因通道而异。为确保客户节点与存储节点之间可以使用备用通道代替故障通道进行数据传输，须使用与故障通道一致的参数建立备用通道。当客户节点监测到上述网络异常时，首先提取故障通道参数，然后使用相应参数建立一个备用通道，见图 2。

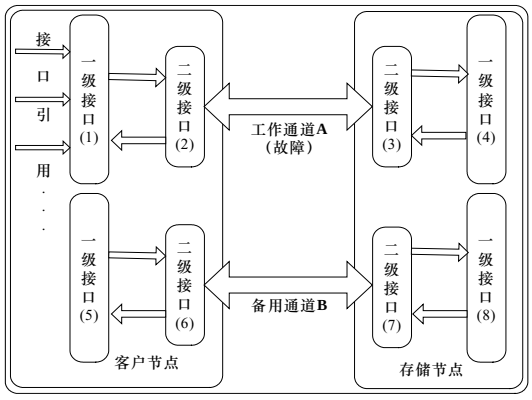

图 2 连接复制示意图

2.2 通道切换

所谓通道切换就是使用备用通道替代发生故障的工作通道进行数据传输。如图 2 所示，BWFS 中客户节点与存储节点之间的数据传输通道包含十分复杂的数据结构：客户节点端提供两级逻辑接口，一级接口与 BWFS 耦合比较紧密，二级接口与操作系统网络协议栈耦合比较紧密，两级接口之间通过众多相互引用(在图 2 中全部抽象为两个方向的引用箭头)建立关联。对于 BWFS 上层而言，一级接口之下的结构是透明的，其不关心一级接口与哪个通道关联，所有对通道的使用均通过对一级接口的引用进行，在客户节点端存在若干对一级接口的无记录引用，无法通过修改这些引用来切换通道；另一方面二级接口之下的部分与操作系统网络协议栈紧密耦合，其无须了解上层结构，通过修改这一部分来切换通道亦不合适。因此只有通过修改一级接口与二级接口之间的关联来实现通道切换。

通道切换是容错过程的关键所在，其难点在于处理一级接口与二级接口之间的复杂关联，切换过程必须确保切换双方每个两级接口之间的相互引用全部按序修正到位，否则可能引起访问无效内存的严重系统错误，见图 3 。切换过程描述如下: (1)将通道 B 的一级接口 5 对二级接口的引用指向通道 A 的二级接口 2; (2)将通道 B 的二级接口 6 对一级接口的引用指向通道 A 的一级接口 1; (3)将通道 A 的一级接口 1 对二级接口的引用指向通道 B 的二级接口 6; (4)将通道 A 的二级接口 2 对一级接口的引用指向通道 B 的一级接口 5。

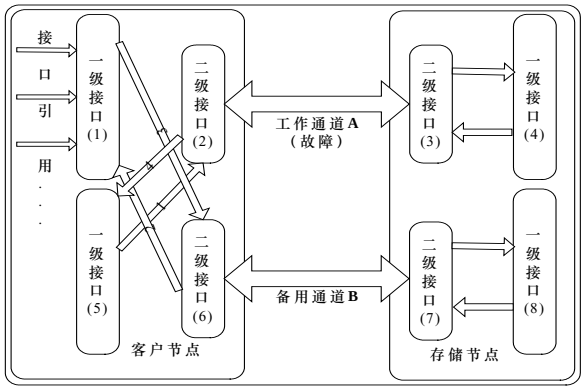

图 3 通道切换示意图

切换过程中，系统暂停处理客户节点与故障通道对应存储节点之间的所有读写请求。待切换处理结束后，原工作通道 A 的一级接口 1 与备用通道 B 的二级接口 6 建立关联，而

备用通道 B 的一级接口 5 则关联原工作通道 A 的二级接口 2。这样从上层看来，系统依旧使用原先的一级接口 1，但实际的工作通道已经切换到备用通道 B，备用通道 B 转变为工作通道，而故障通道 A 连同二级接口 2 可以通过一级接口 5 进行释放。

上述切换过程由客户节点发起，当存储节点发现通道失效(如连接中断或客户节点主动销毁)时，将释放对应通道在本节点所持有的系统资源。

2.3 请求重构

经过连接复制和通道切换处理后，需要对故障通道上尚未处理完毕的请求进行重构工作，用以恢复网络异常时刻故障通道上的请求处理，使之处于可控状态。在 BWFS 中，请求处理状态主要包括下述几种：(1)读请求尚未处理，等待调度；(2)读请求正在发送；(3)读请求已经发出，等待存储节点数据；(4)读请求正在接收数据；(5)写请求尚未处理，等待调度；(6)写请求正在发送；(7)写请求已经发出，等待存储节点确认；(8)写请求正在接收确认。

当系统发生上述网络异常而进行通道切换时，正在使用(状态(2)、(4)、(6)、(80))和即将使用(状态(30)、(7))故障通道进行数据传输的操作将失败，上述状态中只有状态(1)和(5)是可控的，其他状态下的数据是不完整(读请求)或不确定(写请求)的，需要重构。

所谓请求重构是指当发生上述网络异常时，在必要的连接复制和通道切换完成之后，将故障通道上尚未处理结束的、处于非(1)且非(5)状态下的请求按照一定规则重新插入请求队列，等待再次调度，以使该通道恢复到网络异常发生之前某个时刻的可控状态(即该通道上所有请求全部处于状态(1)或(5))。由于请求本身携带绝对位置信息，重复的读写操作不会对系统的数据有效性造成影响(此处暂不考虑多客户节点并发操作所带来的影响)。重构规则如下：

(1)将处于状态(2)(或(6))的请求插入请求队列头部，恢复到状态(1)(或(5))；

(2)将处于状态(3)(或(7))的请求插入请求队列头部，恢复到状态(1)(或(5))；

(3)将处于状态(4)(或(8))的请求插入请求队列头部，恢复到状态(1)(或(5))。

上述优先级依次递减，同一优先级确保其在重构后请求队列中的顺序与重构前的相应顺序一致。经过这样的重构处理，通道可以恢复到网络异常发生之前某个时刻的可控状态，并且保持原系统请求之间的依赖关系，网络异常发生时刻的不完整或不确定数据将通过相应请求的再次调度处理而得到恢复。

3 效能测试

本节针对上述两种网络异常进行相关测试，采用对比方式进行。测试系统包括容错前 BWFS、容错后 BWFS 和标准 NBD 系统。测试系统配置如表 1、表 2 所示。

表 1 BWFS 测试配置

	客户节点	元数据服务节点	存储节点
硬件	数量：1；CPU：Xeon 2.4GHz×2；MEM：1GB；网卡：Intel 82551×1	数量：1；CPU：Xeon 2.8GHz×1；MEM：1GB；网卡：Intel 82551×1	数量：1；CPU：Xeon 2.8GHz×1；MEM：3GB；网卡：Intel(R) 82551×1；存储：3ware 9500 SATA 盘阵(160GB×12，RAID5)
软件	Redhat9.0+BWFS	Redhat8.0+BWFS	Redhat8.0+BWFS

表 2 标准 NBD 系统测试配置

	客户节点	服务器节点
硬件	同 BWFS 客户节点	同 BWFS 存储节点
软件	Redhat9.0+ext3+nbd-2.0	Redhat8.0+ext3+nbd-2.0

3.1 通道读性能骤降化异常测试

各测试系统分别执行一组测试，每组测试不间断循环测试 200 次，每次循环执行先写后读各操作 8GB 数据体。

图 4 给出 3 组测试中足以说明问题的前 60 次循环测试的结果，横轴描述循环次数，纵轴描述读操作 8GB 数据体的耗时。容错后 BWFS 的通道读性能与容错前未出现通道读性能骤降化异常时的性能相当，且在整个测试过程中保持平稳；另外两组测试先后出现通道读性能骤降问题(容错前的 BWFS 在第 50 次循环之后，其通道读性能下降到正常读性能的 15%以下；标准 NBD 系统在第 43 次循环之后，其通道读性能降低为正常读性能的 10%左右)。表明相应技术确实起到容错作用，且未对正常状态下的系统性能产生负面影响。

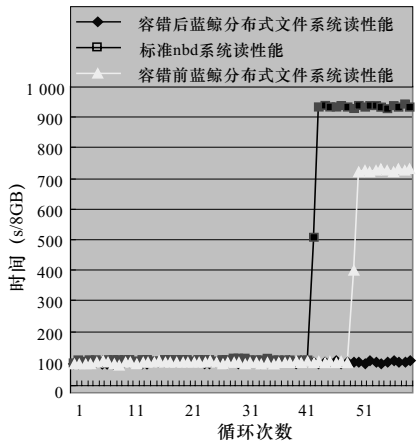

图 4 通道读性能骤降化异常测试结果

3.2 通道连接软故障中断异常测试

测试通过特定扫描进程误操作强制中断客户节点与存储节点之间的相关连接模拟通道连接软故障中断异常。测试方案为：各测试系统分别执行一组测试，每组测试不间断循环测试 100 次，每次循环执行创建文件(1GB)、验证文件的操作，各次循环创建的文件数据具有可识别的不同特征，其间特定扫描进程每 4s 强制中断相关连接一次，确保每次循环中至少出现一次通道连接软故障中断。测试结果如表 3。

表 3 通道连接软故障中断异常测试结果

测试系统	测试结果
容错前 BWFS	I/O 错误，创建文件失败，数据不完整，出现不符合数据特征的错乱数据
容错后 BWFS	循环未受通道连接异常中断影响，创建文件操作成功，文件数据特征吻合
标准 NBD 系统	同容错前 BWFS

容错后的 BWFS 在出现通道连接软故障中断异常情况下实现应用层透明的数据无损传输，系统可用性大为提高；其他未采取相应容错措施的系统则因此异常而导致数据缺失，表明相应技术真正达到容错目的。

4 总结

针对通道读性能骤降化异常和通道连接软故障中断异常，BWFS 采用连接复制、通道切换、请求重构等软件技术

(下转第 86 页)

INT16U killSelf(dataPassToNewTask *localMsg) //任务结束时，//释放任务占用的系统资源

(1)优先级管理：μC/OS-II 中没有提供优先级的动态获得和释放，构造支持多线程的嵌入式 Web 服务器中，如不实现该种动态分配任务优先级的机制，则在处理客户连接而频繁创建服务子进程过程中，μC/OS-II 提供的 56 个优先级将会消耗掉。在本系统中，通过预先设定优先级数组，在创建服务子任务前，调用 getPrio 函数，通过查找优先级数组，判断各优先级使用状态，返回可用优先级；在删除该任务后，调用 ReleasePrio 函数，置该优先级为空闲状态，以释放该优先级。

(2)任务堆栈管理：μC/OS-II 没有提供在删除任务时，删除该任务的堆栈区的功能，应该避免因频繁创建与删除任务而造成无可用堆栈区的情况发生。在本系统中，预先定义任务堆栈内存块数组，由μC/OS-II 的内存管理机制进行任务堆栈区的分配和管理。在新任务创建时，提供给该任务一个堆栈内存块，在删除该任务的同时，将该任务堆栈内存块释放并回收，以便为其他任务创建提供可用的堆栈内存块。实验表明，这样处理可提高系统内存资源的利用率，并有效避免上述情况的发生。

4 嵌入式 HTTP 服务器多线程并发应用实现

根据本项目研究的具体需求，作出嵌入式系统规范，并划分系统软硬件功能，在实时内核μC/OS-II 和 AT91R40008 为软硬件平台核心基础上，以构件的开发模式，完成系统各功能模块的软硬件协同设计。嵌入式 GUI 实现在 LCD 上刷新显示系统运行状态，提供本地用户操作界面的功能；USB 主机模块完成通过 USB 界面采集本地显微图像数据，既可存储到本地的 NAND 型 Flash，也可转存到通用的移动 U 盘，为用户提供文件存储介质的可选性；嵌入式文件系统主要是实现对存储在 NAND 型 Flash 上的数据进行管理，并通过 API 接口为其他应用提供可操作性。

为测试系统网络功能，设置本地 IP 地址为 192.168.0.10，试验局域网网关为 192.168.0.1，子网掩码为 255.255.255.0。程序编译调试通过后，将其下载入目标板中的 NOR 型 Flash，目标板通过以太网线经 Hub 与局域网连接，作为局域网的一台小型嵌入式 HTTP 服务器运行。通过局域网中一台 PC 机，打开 Web 浏览器，在地址栏输入 http://192.168.0.10/ 登录嵌入式 HTTP 服务器，客户端用户在确认身份后，进入操作界面，选择具体的图像操作请求并提交发送给服务器，嵌入式 HTTP 服务器获取用户指令并作相应处理，并返回服务页面。用户可以提交诊断意见给服务器端，并可回显在本地嵌入式系统的用户操作界面上。

本嵌入式 Web 服务器最大可同时为 5 个客户在线请求提供服务，在 10Mbps 局域网内，可以达到 500kbps~600kbps 的实际数据传输速率，可满足通常条件下多用户对大容量数据网络实时传输的需求。

5 结论

在基于实时内核μC/OS-II 和 ARM7Core 的系统板上，结合优化的 TCP/IP 协议栈，构造并实现了针对显微医学远程诊断应用的小型的嵌入式 Web 服务器。本嵌入式 Web 服务器支持经由以太网连入的 Internet，通过类 BSD 套接口与协议栈交互数据，绑定 80 号端口，监听并处理多客户机连接，服务进程作为μC/OS-II 内核管理下的实时任务形式运行。客户端通过普通浏览器访问和控制本嵌入式系统，简化了使用难度，达到了预期的功能。基于软硬件平台的嵌入式 Web 服务器构造，便于复杂系统的网络模块构件化实现，缩短了开发的时间，同时提高了系统的可靠性、可行性。

参考文献

1 吕京建，张宏韬. 基于嵌入式中间件的系统开发方法[C]. 第 13 届中国微计算机年会，北京，2002: 414-419.

2 Cai X, Michael R L, Kamfai W. Component-based Embedded Software Engineering: Development Framework, Quality Assurance and Generic Assessment Environment[J]. Software Engineering and Knowledge Engineering, 2002, 27(2): 107-133

3 彭少熙，孙政顺，杜继宏. 家庭网络中的嵌入式 Internet 方案[J]. 电子技术应用，2001, 27(10): 47-50.

4 Labrosse J J. 邵贝贝译. μC/OS－II——源码公开的实时嵌入式操作系统[M]. 广州：中国电力出版社，2001

5 Grasic B, Mlakar P. Use of Open Source Operating System and TCP/IP Connectivity in Urban Environmental Monitoring[C]. Proc. of the 11th International Conference on Advanced Thermal Processing, Maribor, Slovenia, 2003: 1257-1261.

~~~~~~

(上接第 74 页)

加以容错，其技术是可行的，其性能是可以接受的。实测表明针对特定的网络故障，相应的软件技术在无额外硬件支持的情况下发挥了预期的容错功能，确保了系统运行的性能平稳，实现了应用层透明的数据无损传输，提高了 BWFS 的可用性。网络容错是复杂的系统问题，对于类似网络驱动崩溃、网络连接硬中断等异常，BWFS 现有的容错软件技术不能确保应用层透明的数据无损传输，需要配合以硬件容错措施。如何在网络异常情况下确保数据的无损传输及存储节点的数据有效性是值得深入研究的课题。

参考文献

1 Farley M. 孙功星，蒋文保，范　勇等译. SAN 存储区域网络(第 2 版) [M]. 北京：机械工业出版社，2002-04.

2 Stevens W R. 范建华，胥光辉，张　涛等译. TCP/IP 详解，卷 1：协议[M]. 北京：机械工业出版社，2000-04.

3 LeMahieu P, Bohossian V, Bruck J. Fault-tolerant Switched Local Area Networks[C]. Proc. of the First Merged Int. Parallel Processing Symposium and Symposium on Parallel and Distributed Processing, 1998: 747-751.
~~~~~~