电信级操作系统CGEL中的高可用性设计

王继刚[1,2] 郑纬民[1] 谢世波[2] 钟卫东[2]
(清华大学计算机科学与技术系 北京 100084)[1] (中兴通讯中心研究院成都研究所 成都 610041)[2]

摘 要 高可用性对于电信设备的持续运行至关重要。当软硬件出现故障时,高可用性技术会帮助系统保持正常运行。为了增强错误面前电信设备的稳定性,通过深入分析电信设备的高可用性需求,基于CGEL(电信级嵌入式Linux;Carrier Grade Embedded Linux),在运行监控、错误控制、故障转移、在线升级等方面进行了一系列高可用性设计。这些设计功能独立,且有机结合,能够帮助电信设备有效地抵御各类软硬件故障,保证系统不停机平稳运行。

关键词 高可用性,电信级,状态监控,故障转移,冗余
中图法分类号 TP911.22 **文献标识码** A

Design of High Availability in Carrier Grade Operating System CGEL

WANG Ji-gang[1,2] ZHENG Wei-min[1] XIE Shi-bo[2] ZHONG Wei-dong[2]
(Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)[1]
(Central Research Institute Chengdu Graduate School, ZTE Corporation, Chengdu 610041, China)[2]

Abstract High availability today plays an important role in protecting continuous running of critical telecommunication equipments. When the hardware or the software goes wrong, high availability technologies can help the system to maintain normal. In order to improve the robustness of telecommunication equipments in the presence of errors, this paper analysed the high availability requirements of telecommunication equipments. Based on carrier grade operating system-CGEL (Carrier Grade Embedded Linux), this paper proposed and implemented a series of high availability designs, including of condition monitoring; error control; failover; on-line operations. The functions of these high availability designs are relative independent, but closely related each other. The experimental results show that these designs can help telecom equipments to effectively resist the hardware and software failures, and ensure non-stop telecom system for stable operation.

Keywords High availability, Carrier grade, Condition monitoring, Failover, Redundancy

随着网络通讯技术的快速发展,各种新兴的电信业务,包括VOIP、可视电话、流媒体等也在迅猛增长。为了满足这些新的语音和数据业务需求,电信行业正在发生转变。今天的用户不仅需要高速宽带的通信网络,更需要连续不间断的服务传递,这就对各类电信设备提出了严格的要求:在提供可预测的性能和良好的通信质量的同时,还要保证服务高可用性[1](即设备正常运行时间超过99.999%),维持网络平稳运行。

目前,国内外电信设备设计者们主要考虑通过冗余配置来提高系统的可用性,以防硬件出现故障。然而,硬件冗余只是保证高可用性的一个方面。随着系统越来越复杂,由硬件故障引起的停机比重正在逐渐下降,如表1所列。从业界对电子交换系统和事物处理系统不可用原因的调查结果[2]中可以看出,获得系统高可用性更多地受到了软件和流程等因素的影响。仅仅考虑硬件故障来提高系统可用性,对于真实环境而言是不具代表性的。

表1 系统不可用原因分布

停机的原因	电子交换(%)	事物处理(%)
硬件故障;环境	20	45
软件故障;不正确的恢复	50	30
不正确的流程	30	20
其它,比如升级	包含在前面数值内	5

操作系统作为软件的控制核心,它的任务处理能力及稳定性直接影响电信设备的整体性能和高可用性。为了满足当前电信设备的各方面需求,本文提出了一种电信级Linux操作系统CGEL[3](Carrier Grade Embedded Linux),并根据影响系统可用性的多种因素,在CGEL中提出了一系列增强系统可用性的相关设计,这些设计能够使CGEL有效抵御软件或硬件故障,具备软件在线升级能力,保证电信系统不停机平稳运行。

1 电信设备高可用性需求

电信用户期待着他们的语音和数据服务总是可用的。整

到稿日期:2009-02-05 返修日期:2009-04-21 本文受863国家重点基金项目(2004AA1Z2351),电子信息产业发展基金项目(XDJZ-0412-03)资助。

王继刚(1978—),男,博士后,主要研究方向为集群系统、可信计算等,E-mail:gloryjgwang@gmail.com;郑纬民(1946—),男,教授,主要研究方向为网格计算等;谢世波(1973—),男,博士生,主要研究方向为计算机通讯网络等;钟卫东(1972—),男,硕士,主要研究方向为操作系统等。

个电信网络的可用性取决于网络中单个系统的可用性。为了保证24小时不间断服务，系统必须能够在运行的电信网络和服务器上完成自身的维护和扩展[4]，同时不中断正在执行的服务；必须能够抵挡部件故障，设置部件冗余。软件故障也可能对系统的可用性造成巨大影响，完善的软件错误处理机制对单个系统可用性而言是必需的。为了防止单点故障，可用性还可以通过集群加以增强。总的说来，系统在可用性方面的需求包括在线操作、冗余、监视、健壮软件等。

当系统中的软件或硬件被替换或升级时，在线操作能够使系统持续提供服务。比如，当一个文件系统需要修复，修复过程可能要求重启系统。但是，电信设备要求强制卸载文件系统是可能的，同时允许无重启的修复和重新加载。在不关闭节点和网络的前提下，对硬件进行替换和升级的能力可以大大提高服务可用性。

一个高可用系统必须由冗余部件组成，同时要能够有效利用这些冗余硬件，保证有部件失效时系统仍能持续运行[5]。在理想的情况下，可用性设计可以从系统中消除所有单点故障。利用冗余的电信设备，比如主备节点，与网络故障恢复软件结合在一起，能够有效提高系统可用性。内存部件的冗余难以实现，但错误检测和修复可以用于解决内存单元故障。电信级操作系统需要软件错误纠正代码支持，硬件中的单个位错误会被内核检测报告并在日志中记录。当检测到不正确的多位错误时，内核会启动异常处理流程[6]。

硬件和软件故障的快速检测需要运行监视[7]，运行监视还需要检查那些可能要失效的软硬件，比如ECC内存检查、硬盘预测性分析、在预测方式中无法响应的进程。系统运行监视需求中的实例包括进程非介入监视和内存过载处理。进程非介入监视检测进程的非正常行为，比如进程死亡或创建一个新进程。内存过载处理则监视系统内存的使用率，当内存使用率超过指定阈值时控制进程行为。

健壮软件不仅指操作系统软件、中间件、应用软件的高质量，还包括在不关闭系统的前提下维护和升级软件的能力[8]，在许多情况下，能够维护连续的服务可用性。一个动态补丁需求能够在进程不中断情况下修改进程，通过在进程中不同点处设置CPU使用阈，一个额外CPU循环检测的需求可以检测异常进程行为，捕获比如死循环或颠簸等问题，并进行处理，比如重启进程。

2 电信级操作系统CGEL

为了满足电信网络特殊的应用需求，在深入研究Linux优缺点的基础上，设计并开发出一种新型的电信级操作系统——CGEL。其设计核心是在不改变Linux的系统构架的基础上，将实时应用从用户态移植到内核态，并在内核中为应用程序提供实时支持，满足实时应用在多任务管理、互斥、通讯等方面的需求；同时，完善内核调度机制，使之具有实时与非实时任务混合调度能力；CGEL在内核态和用户态均提供了标准的C/C++库接口，能够帮助用户快速方便地进行双态应用开发。最后，为保证系统及应用能够在电信环境下稳定运行，CGEL提出并实现了一系列高可用性设计。

如图1所示，CGEL是在标准Linux内核基础上添加了满足电信设备需求的性能增强模块。比如线程增强模块KTH提供了实时线程管理能力，以支持实用应用在内核运行；进程间通信增强模块KIPC提供了用户态进程间的快速消息能力，也支持内核态和用户态应用间的通信；工程管理模块通过模块化Linux板级支持包，加强了CGEL工程开发能力。

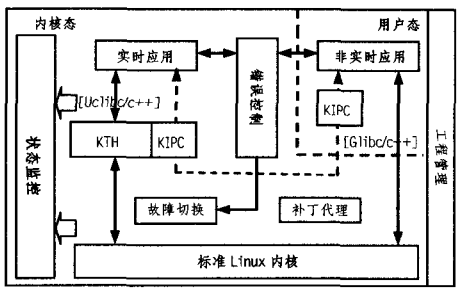

图1 CGEL内核体系结构

在提升系统可靠性方面，CGEL根据电信设备高可用需求，从多角度进行了增强性设计。状态监控模块动态收集内核及应用的运行数据，检测系统运行异常，并提供相关信息供上层应用和用户查看；在遇到应用错误或系统异常时，错误控制模块提供各种错误处理机制，帮助系统或应用快速容错恢复；故障转移模块基于设备冗余，能够在系统出现重大软硬件故障且不可修复时，高效透明地进行故障转移，保证应用的持续运行；动态补丁模块则支持系统在运行状态下对应用进行动态升级修改，提升系统的可用性。这些模块功能相对独立，之间又紧密联系，共同组成了一个有机的整体，保证了CGEL系统运行的稳定性和可维护性。同时，状态监控、错误控制等模块还提供了丰富的标准C/C++语言接口，方便用户挂接一些自定义的可用性功能，增强了整个系统的灵活性和可扩展性。

3 CGEL高可用性设计

3.1 状态监控

状态监控模块提供了系统及应用的运行监控功能，支持用户自定义监控功能的注册调用，通过主动或被动的方式，对系统及应用的运行状态进行实时监控和保存，反馈状态信息，捕获错误异常。状态监控是整个高可用性设计的核心，它与错误控制、故障转移等模块紧密配合，组成全系统的异常监控和故障处理机制，确保系统正常稳定运行。

CGEL状态监控的设计思想是通过统一的手段获取和控制系统及应用的运行状态，还要具有一定的灵活性。为满足上述目标，本文以层次化的方式来设计状态监控模块，如图2所示。

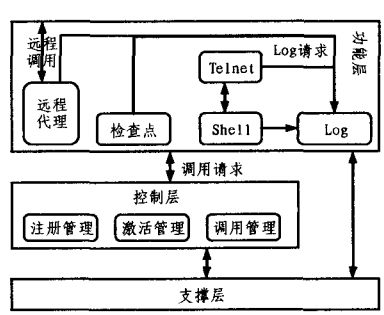

图2 状态监控模块层次结构

整个状态监控模块自下而上分别为支撑层、控制层、功能层。支撑层提供了各项基本的监控功能，包括系统和应用的

信息收集、状态检测保存、错误异常捕获以及状态信息查看，这些具体功能是整个状态监控模块的基础。支撑层上面是控制层，它提供了一个灵活高效的监控功能管理机制。控制层由3部分组成，分别是监控功能的注册管理、激活管理以及调用管理。通过对监控功能信息、激活点信息和激活信息的维护，完成应用对监控功能的激活请求和调用请求，实现模块对各类监控功能的管理。同时，控制层提供了统一的调用接口，使上层应用能以统一的方式调用各项运行监控功能。功能层则实现了一些成型的高级状态监控手段，比如Shell系统监控命令、检查点、系统运行日志、远端代理等，用户可以直接使用这些功能，方便应用开发。

通过层次化的划分，可以将监控功能按照不同的能力和通用性划分到不同的层次中，以更好地满足不同应用对状态监控功能的要求。

3.2 错误控制

针对状态监控捕获到的系统异常，错误控制模块不仅提供了完善的错误处理机制，还提供了用户可自定义的异常处理接口，方便用户扩展错误处理的功能和范围。错误控制模块包括错误框架和错误处理两个子功能模块。错误框架是一个逻辑的概念，负责维护错误处理状态机模型，控制着整个处理流程，使程序错误处理能够有序地进行；系统或应用的错误异常被运行监控模块捕获后，由错误处理部分进行统一处理。对捕捉到的错误可以提供基本处理流程，也支持用户自定义处理手段。同时，错误控制模块要调用运行监控模块的其它服务进行错误信息的存储和告警操作，这是对错误定位必不可少的要求。

针对CGEL应用能够在用户态和内核态同时运行的情况，错误控制也提供了双态运行方式。图3是错误控制在用户态和内核态环境下的运行方式，重点描述了状态监控与错误处理之间的关系。其中状态监控的错误异常捕获是核心，所有异常都会进入，它将调用错误处理，错误处理分析用户设定的错误处理策略；同时，错误控制模块可以调用状态监控提供的其它服务进行信息报告、状态保存等操作。错误处理在内核态和用户态同时提供。在用户态的处理机制，通过信号进行驱动，完成单个进程的错误管理功能。而内核态的错误处理机制，可以同时接受错误捕捉机制的调用，完成对系统的错误控制功能。

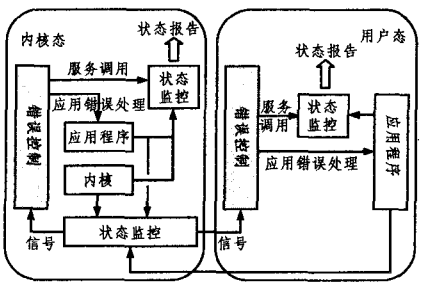

图3 双态下错误控制运行机制

3.3 故障转移

在设备冗余的环境下，CGEL提供了故障转移功能。当系统出现难以恢复的软硬件故障时，故障转移模块能够快速透明地进行设备切换，保证电信设备业务不间断运行。

故障转移包括应用状态的转移（系统中各服务应用的数据同步）和网络连接状态转移（与客户端的网络连接切换）。触发故障转移条件有主动转移和被动转移两种，转移的设计原则是快速和透明。故障转移的工作涉及到两个方面，分别由应用控制和通信控制两个子功能模块完成。应用控制模块负责应用状态转移，控制所有应用进程或线程（包括通信控制模块）的转移顺序和数据同步；通信控制模块运行负责底层网络连接切换和通信链路上缓存的消息处理。在故障转移过程中，应用控制模块和通信控制模块进行协调，按照分工分别完成应用的状态转移和网络连接切换。同时，主备节点对客户端只提供一个IP地址，绑定在主节点上。当故障转移时，原主节点解除对该IP地址的绑定，由新主节点将该IP地址与自己MAC地址绑定起来。建立IP地址与MAC地址的映射关系，是为了系统内部节点能正确区别主备节点。故障转移模块的体系结构如图4所示。

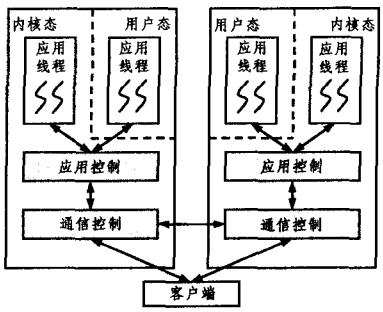

图4 故障转移模型

主动故障转移过程如下：主节点应用控制模块通知通信控制模块，故障转移开始。主节点收到的客户消息不再向上层应用派发，而是直接转发到备节点缓存。同时，停止向客户端发送服务数据，并将未传输完的服务数据转发到备节点缓存，等到故障转移结束后重新发送。在通信控制模块处理网络连接转移的同时，应用控制模块负责其它应用进行故障转移，各应用停止处理业务，保存当前状态。然后，主节点应用控制模块通知备节点进行主备角色转换。在这一过程中，主节点将各应用状态的检查点同步保存到备节点。主备角色转换完成后，备节点成为新主节点，开始继续处理业务。其通信控制模块将缓存的客户消息派发至应用进程处理，并通知客户端，故障转移结束，恢复服务交付。

网络连接状态切换在通信控制模块的支持下完成。故障转移开始，通信控制模块先设置系统状态为故障转移状态，并通知相关的客户端，停止转发接收到的网络消息，开始向备节点同步传送已经稳定了的状态数据。在转移期间把收到的客户消息转发到备节点，备节点将所有转发的客户消息缓存到消息队列中。接着是网络链路切换，通信控制进程将当前主节点与客户端的网络连接状态发往备节点，备节点完成自己网络连接状态的更新，根据网络连接状态的信息向客户端发送故障转移结束通知，作为新主节点开始向各应用派发消息。客户端收到倒换结束的消息后，恢复自己的状态为正常状态，发送缓冲在自己发送队列中的消息。

系统在正常运行过程中，主备节点的状态监控模块之间保持联系。一旦发现主节点失效或在指定时间阈内无任何反应，备节点将启动被动故障转移流程，承担主节点的角色。步骤如下：备节点应用控制模块向其通信控制模块发出故障转移通知，通信控制模块随即通知客户端缓存发往主节点的消息。在备节点通信控制进程处理链路切换的同时，备节点应

用控制模块负责应用的主备转换，首先激活应用程序，并通过保存的检查点和消息日志进行状态恢复。在被动故障转移中，应用程序不进行主备之间的数据同步缓存及转发。当恢复各应用程序状态完成之后，新主节点应用控制模块通知通信控制模块应用程序主备转换完成。通信控制模块向客户端发出链路切换命令，网络连接恢复正常。

3.4 在线升级

动态补丁代理负责在不中断应用的前提下对其进行修改，保证应用不间断运行。补丁代理通过接受并解析控制命令，根据补丁文件信息进行相关操作，完成应用补丁的加载与卸载。同时，补丁代理与状态监控模块协作，支持远端和本地的补丁控制及异常处理，增强用户操作和系统维护。

图 5 展示了补丁代理模块结构及与其它模块间的关系。可以看出，补丁代理模块在状态监控和错误控制模块的支持下运行。补丁代理由两个部分组成：补丁控制处理补丁的运行工作，补丁操作则进行具体的运行代码补丁操作。

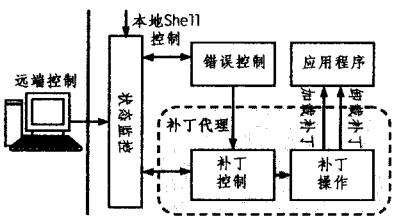

图 5 补丁代理模块功能结构

通过状态监控模块的调用代理和 Shell 功能，补丁控制可通过本地命令行或远端界面等方式接收用户指令，读取补丁信息文件数据，调用补丁操作。同时，向状态监控模块反馈状态信息，便于用户查看。补丁控制还与错误控制模块配合，当系统运行出现异常时，检查分析异常是否由于新加载的补丁所引起，并在必要时卸载补丁进行修复。

补丁控制部分还负责维护补丁文件。补丁文件是一个将需要修补的函数集中在一起的自定义格式文件；补丁模块对应于每一个被补文件的对象文件源；补丁函数是单一的更新函数。如图 6 所示，补丁文件信息以层次关系存放，每个补丁文件下面分为多个补丁模块，而每个补丁模块则由多个补丁函数组成，补丁文件之间以链表结构相连。这种层次关系也反映了补丁文件在内存中的存放格式。补丁有 4 个基本状态：加载、激活、去激活、卸载。为了提高补丁控制的灵活性，补丁激活、去激活和补丁状态查询可以在文件级和函数级两个层次上进行。补丁文件加载在用户态下采用共享库的方式实现，对于动态符号的查找采用系统提供的动态库方式。在内核态下被补丁程序采用动态插入的方式，补丁中符号的重定位则采用了内核模块的符号搜索机制。

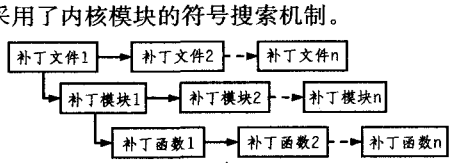

图 6 补丁信息存放格式

补丁操作完成具体的补丁操作，包括用新代码替换原有代码以及恢复原有代码动作。代码替换以函数为单位进行，通过获取到原有函数代码的入口，将其替换为一个长跳转指令，跳转到新函数实现中，即完成了补丁激活。同时，在代码替换时保存原有代码，在补丁去激活操作中将保存的原有代码写回去，这样原有代码又可继续工作。

结束语 为了保证电信设备持续运行，为用户提供不间断的业务服务，本文通过分析电信设备的高可用性需求，在电信级操作系统 CGEL 上进行了一系列设计，提出并实现了由状态监控、错误控制、故障转移、在线升级等方面组成的高可用性解决方案。这些设计功能相对独立，联系紧密，共同组成了一个有机整体，大大提高了 CGEL 系统运行的稳定性和可维护性。目前，CGEL 已经成功应用于网络 ADSL、数据路由器等多款电信设备上，运行稳定，取得了良好的效果。实践表明，基于高可用性设计的 CGEL 能够有效抵御软硬件故障，具有应用在线升级能力，保证电信系统长时间平稳运行。

构建高可用性的计算系统存在许多困难，但它对提高系统可依赖性和可维护性具有不可估量的巨大作用。针对如何提供系统高可用性，近年来国际上的关注点正在由硬件向软件和流程转移，虽然取得了一定进展，但离最终目标仍有相当大的差距，面临着很多问题与挑战：通过数学方法精确描述和规范高可用性的设计与实现；对故障本质的研究，更精确、更细粒度的故障定义，更细级别的性能监控；智能化自动诊断和错误处理技术的研究，以及面向可用性的性能测试。这些都是很具有挑战性的研究课题，有待进一步研究。

参 考 文 献

[1] The Service Availability Forum. Application Interface Specification[EB/OL]. http://www.saforum.org, April 2003

[2] Laprie J C, Kanoun K. X-ware Reliability and Availability Modeling[J]. IEEE Transactions on Software Engineering, 1992, 18(2): 130-147

[3] 王继刚，顾国昌，徐立峰，等. 强实时性 Linux 内核的研究与设计[J]. 系统工程与电子技术，2006，28(12)：1932-1935

[4] The stratus Active Service architecture : Remote access to mission-critical support, 24/7 [R]. Stratus Technologies, 2004

[5] Li M, Goldberg D, Tao W, et al. Fault-tolerant cluster management for reliable high-performance computing [C]// International Conference on Parallel and Distributed Computing and Systems. Anaheim, CA, USA: IEEE Press, 2001: 480-485

[6] Lemos R, Romanovsky A. Exception Handling in the Software Lifecycle[J]. International Journal of Computer Systems Science and Engineering, 2001, 16(2): 167-181

[7] Leangsuksun C, Liu T, Shen W, et al. Building high availability and performance clusters with HA-OSCAR toolkits [C]// Proceedings of the High Availability and Performance Workshop. Santa Fe, NM, USA: IEEE Press, October 2003: 597-606

[8] Hicks M, Nettles S. Dynamic software updating [J]. ACM Transactions on Programming Languages and Systems (TOPLAS), 2005, 27(6): 1049-1096