采用分布式实时操作系统的容错系统的设计与实现

戈应安　刘松强　王砚方

（中国科技大学近代物理系快电子学实验室，合肥 230027）

介绍了设计和实现的一个分布式容错系统的实验原型。系统采用 PC 双机备份结构和 QNX 分布式实时操作系统，通过用监视器进程定时检测工作节点状态和主机和备份机相互切换的方式，实现对单节点静默失效和偶发故障的实时容错。由于系统的容错功能是采用软件方式实现，使得该系统不仅开发成本低，而且功能易于扩充和升级。

关键词：分布式系统　容错　实时

0　引言

分布式操作系统与网络操作系统虽然都使用网络，但是两者具有不同的结构和工作方式。网络操作系统按客户－服务器方式工作，系统的客户部分和服务器部分各自设置在不同的节点上，它是一种各个组成单元分布在网络各节点上的集中式操作系统。分布式操作系统通常是按对等方式下工作的，客户和服务器可以分处、也可以共处在系统的任意节点。分布式操作系统有以下两个基本特征：(1)分离性，即系统内有大量独立、自治的和可以相互通信的处理机节点；(2)透明性，对用户和编程者而言，整个系统如同一个单一处理机的系统。透明性是分布式系统最重要的特征，它极大地简化了应用软件设计。

传统的容错设计大多依靠硬件冗余设计，要采用各种专用芯片、特殊电路和专门的实现手段，并投入大量的研究资金和开发时间。这使得成熟的容错设计总是落后于现代技术的发展步伐，其系统性能往往远低于非容错系统。这使用户在选择系统时不得不在系统性能与容错要求之间作出妥协。

随着微电子学技术的不断发展，芯片集成度迅速提高，价格持续下降，部件的出错概率大大降低，这使得电路级与板级的硬件冗余的必要性大为减弱。由于分布式系统的每个自治的节点形成独立的故障隔离区，它可以很方便地实现节点级的冗余备份。因此，可以采用分布式操作系统，将分布与备份相结合来获得系统容错能力，组成分布式容错系统。这种基于软件的系统易于扩充容错能力和处理能力，且不必改变其结构，仅通过对构成节点的计算机的升级就可提高系统性能，从而可以充分利用现代成熟的通用计算机的计算能力，跟上更新换代的步伐。

1　分布式容错实验系统

1.1　分布式容错基本原理

分布式系统故障的最常见形式是静默失效(fail silent)，这时出故障节点的处理机不再响

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.c

应任何输入，也不再产生任何输出。单节点静默失效可分为两种情况：主节点的静默失效与备份节点的静默失效。在分布式容错系统中，可以采用监视器进程检测主机的控制流的方法来发现节点的静默失效。

对于主节点的静默失效情况，当备份节点的监视器进程 Watchdog 发现主节点失效时，就向系统管理员发出失效警告，同时运行主节点备份进程从而升格为主节点。在对失效节点排除故障重新启动后，原主节点就成为备份节点并运行监视进程；新的主节点的服务器进程 MP 收到新的备份节点监视进程发来的消息后，就终止本节点的监视进程，从而使系统恢复到失效前的正常工作状态。对于失效时未完成的客户服务，原主节点停止执行，在备份进程启动后再接受客户请求。对备份节点的静默失效，可以采用由主节点的服务器进程 MP 定时检测备份节点的方法。发现其失效后，在本地节点运行监视进程，同时向管理员发出备份节点失效的警告。其后的恢复过程同主节点失效一样。

网络的局部失效等同于某一个节点的静默失效，任一节点在发觉网络联络中断时（此时其它节点无法访问该节点）均终止当前进程并向管理员发出网络故障警告。由网络管理员在修复其联接后重新启动该节点，若此节点为服务节点，则过程如前文所述；若为普通的客户节点，则由客户决定其运行状态。

1.2 系统的结构

作为对系统分布容错的初步研究，设计了由主机和备份机组成的 PC 双机备份服务器，二者通过以太网卡连接到 HUB 上，与各个客户机一起构成分布式容错系统。除了运行客户进程的客户机外，用作服务器的两台机器也可运行客户进程。系统的结构如图 1 所示。

操作系统采用 QNX（V4.23A 版）分布式多任务实时操作系统[1]，该操作系统专门用于 Intel 的 80×86 系列芯片和 PC 体系结构，采用消息传递作为进程间通信的基本手段，具有网络容错能力，软件开发使用 QNX 专用的 Watcom C（V10.6 版）编译器[2]。

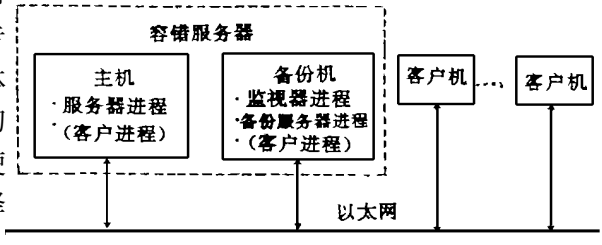

图 1 实验分布式容错系统的结构

系统设计采用被动式的主机/备份机运行模式：主机运行服务器进程，响应客户进程的服务请求，完成各项服务；备份机运行监视进程与备份服务器进程，监视主机的运行状况，在主机完成某个服务请求时备份主机的相应工作状态。当主机由于掉电或其它故障造成失效时，备份机用该机的服务器进程接替原来主机上的服务器进程的工作，并发出主机失效的报告。失效节点在消除故障重新工作后，自动充任备份机，运行监视进程，向此时的主机发出监视进程运行的消息；主机收到该消息后，停止本节点监视进程的工作，恢复双机备份工作模式。客户进程无需了解提供服务的服务器进程所在节点，服务器在分布式网络上任一节点运行。

上述设计中，主节点的服务器进程 MP 与备份节点上的监视器进程 Watchdog 之间的消息传递给系统带来了额外的开销，这包括 CPU 的开销和网络负载的开销。对于 CPU 开销问题，可以通过调节监视器进程的节点失效检测信息传递的频度，在用户对工作性能的要求与对失效延迟响应的承受能力之间折中。频繁的检测可以迅速察觉失效的发生，但会占用较多的 CPU 时间，从而降低系统性能；较低的检测频率则会使系统在发生失效时反应时间较长，使用

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.c

户承受较长的恢复等待时间。对于网络负载问题，由于 QNX 支持节点间多网线连接，可以简单地通过增加网线来增大网络带宽，可在主节点与备份节点间增加一条网线(和相应的网卡)专门用于失效检测信息传递，这样既提高了两机间通讯效率，又不占用公共网络的带宽。

1.3 软件设计

为达到双机备份的目的，在实验系统运行以下进程：服务器进程 MP，工作代理进程 Worker，计时器进程 Timer，监视进程 Watchdog，客户进程 Client。它们分别运行在主机和备份机结点。

1)主机上运行的进程

服务器进程 MP：运行时，首先查询网络的名称服务，若未发现相同的服务器进程，便注册全局名 MP-SERVER；然后调入工作代理进程、计时器进程与监视进程；接收客户进程 Client 发出的服务请求，查看可提供资源，在许可的条件下为其分配一个工作代理进程；定时触发监视进程 Watchdog 的一个定时 Proxy。图 2 所示为服务器进程的流程图。

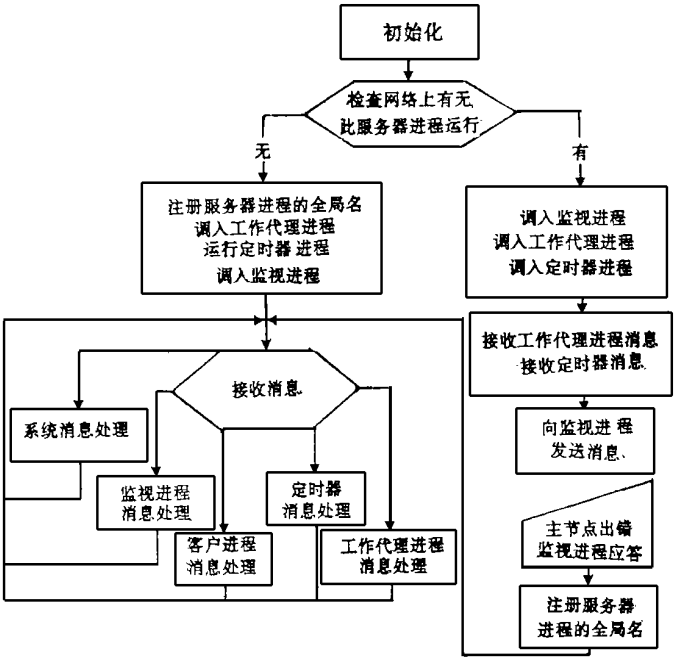

图 2　服务器进程流程图

工作代理进程 Worker：运行后向服务器进程 MP 发出准备好消息，等待分配任务；得到任务后进行服务操作并完成服务；然后向服务器进程 MP 发出报告完成及结果的消息，等待下一次任务分配。

计时器进程 Timer：为服务器进程 MP 提供定时服务。

2)备份机上运行的进程

备份服务器进程 MP _ R：结构与主机上的服务器进程相同，查询网络发现已有全局名为 MP _SERVER 的服务器进程在运行，使进程进入备份状态，也调入工作代理进程、计时器进程与监视进程，等待接收监视进程的消息；在获得主机节点失效的消息后，注册全局名 MP _SERVER，成为服务器进程。

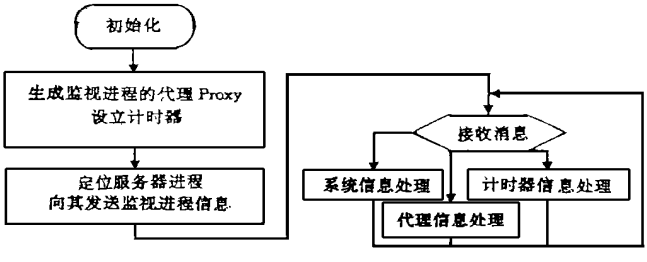

图 3　监视器进程流程图

监视器进程 Watchdog：接收服务器进程 MP 的定时消息，在发现服务器进程 MP 的定时消息超时时，检测其所在节点的状态；在主机节点发生失效时，使本节点的备份服务器进程 MP _ R 成为全局名为 MP _SERVER 的服务器进程。图 3 所示为监视器进程的流程图。

工作代理进程 Worker 和计时器进程 Timer：同主节点。

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.c

3)可在网络上任一节点运行的进程：

客户进程 Client：提出服务请求，等待结果答复；若应答信息有误，则重新定位服务器进程并重发服务请求。

1.4 对基本系统的改进

上节所描述的实验系统是被动地对主机进行备份，这在有些情况下无法采用，需要对系统的工作模式进行改进，使其成为一种主动备份的工作方式，即主机的服务器进程 MP 每接收到一个服务请求，先采用非阻塞式消息传递方式通知备份服务器，在得到应答后才进行服务，并在确认备份服务器任务完成后才通知客户进程。

为了防止工作过程中主节点的某些偶发故障，我们可以使监视器进程 Watchdog 也通过注册获得全局名称 WD，并针对工作代理进程采用签名确认的方式，避免由于偶发故障造成系统出错。为简单起见，对主节点的某些偶发故障引起的工作代理进程 Worker 失效，在实现容错时采用了由服务器进程 MP 终止该失效进程，并重新为客户请求分配一个工作代理进程的方式；而不是将容错步骤设置在工作进程 Worker 的内部，这样就简化了工作代理进程 Worker 的设计。

QNX 系统本身提供的对网络上节点失效定时检测的系统函数以约 4s/次的频度运行，响应时间过长且无法灵活调整。为此，在 MP 服务器进程与监视进程 Watchdog 内部设置定时器，可以根据应用系统所能容忍的最大响应延迟进行调节，以缩短节点失效检测时间。我们在两个节点同时运行 MP 服务器进程与监视进程 Watchdog，形成两个功能完全对称的节点，将其中一个置为备份进程，处于悬置状态，仅在有节点失效情况下才投入运行。这样就省去了将进程调入内存所需花费的时间，仅需要考虑节点失效检测时间、全局名建立时间和进程调度时间。

作了以上改进后，各进程功能如下。

服务器进程 MP：查询网络的名称服务后未发现相同的服务器进程，注册全局名 MP_SERVER；调入工作代理进程、计时器进程与监视进程；接收客户进程 Client 发出的服务请求，查看可分配资源，在许可的条件下为其分配一个工作代理进程，并触发监视进程 Watchdog 的一个相应 Proxy，通知其获取客户信息；接收监视进程的消息并分别进行处理：获取客户信息或工作进程出错；定时触发监视进程 Watchdog 的一个定时 Proxy。

工作代理进程 Worker：运行后向服务器进程 MP 发出准备好消息，等待分配任务；得到任务后进行服务操作，完成服务；结束后向服务器进程 MP 发出结果与完成消息，等待下一次任务分配。在向服务器进程 MP 报告工作结束前，向监视进程 Watchdog 发送预先设定好的签名消息，等待其确认，以保证程序流程无误或运行结果合理。只有在被确认后，才结束操作，向服务器进程发送结果。

监视进程 Watchdog：接收服务器进程 MP 的定时消息，在发现服务器进程 MP 的定时消息超时时，检测其所在节点的状态；在主机节点发生失效时，使本节点的备份服务器进程 MP_R 成为全局名为 MP_SERVER 的服务器进程；接收主节点工作代理进程的签名消息，进行合理性校验或正确性检查，向通过检测的工作进程发回确认消息；未通过检测的则报告服务器进程。

客户进程 Client：提出服务请求，等待结果答复；若应答信息有误，则重新悬置本进程 1s 后重新定位服务器进程并重发服务请求。

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.c

计时器进程 Timer 没有做改变。

1.5 系统测试结果

实验系统所使用的主机为主频 133MHz、内存 32MB 的 Pentium 机，备份机为主频 45MHz、内存 8MB 的 486 机。实测，一个 4 字节的消息在两机之间传递的时间为 2.5ms，在同一机器上进程间传递一个 4 字节消息的时间分别为 16μs(Pentium)和 110μs(486)。在实验中我们所设计的工作代理进程的计算时间为 10ms。由于进程较少而且简单，系统进程调度所用的时间开销可以忽略。

对于我们所设计的客户服务，在备份机(486)不运行监视器进程 Watchdog 时，Pentium 机上运行的服务器进程 MP 完成一次本地发出的客户请求需时约 11ms，完成一次从备份机发出的客户请求需时约 62ms。当在 486 上运行监视进程 Watchdog 时，完成一次本地发出的客户请求需时约 14ms，完成一次从备份机发出的客户请求需时约 64ms。由此可以看出：由于监视器进程的引入，主机对本地客户服务每次增加了 3ms 开销，对远程客户服务每次增加 2ms 开销。

监视进程若超过 200ms 未接收到主节点服务器进程的定时消息，则对主节点进行检测。若检测结果为主节点失效，则应答本节点的备份服务器进程，使备份升格为主服务器进程。检测主节点失效所需的最长时间为主节点自发出定时消息后即失效，至备份服务器进程接收到应答消息，实验中测得约 250ms。备份申请全局名至其在网络生效所需的最大时间可以由系统设定，实验中定为 1s。所以在客户进程得到系统返回的出错信息后，采用悬置进程 1s 后再定位服务器进程的方式以保证备份服务器可靠升格为主服务器进程。

实验中采用切断机器电源、终止服务器进程或切断网络连接的方式引入节点的静默失效，用伪随机发生器随机发出错误签名的方式引入工作代理进程的偶发错误，客户服务(仅指在未失效节点运行的客户进程)都没有异常，只是在恢复期间的响应延迟稍有增大。偶发失效的平均恢复时间为 7ms，最大为 11ms；静默失效的平均恢复时间为 1.2s，最大值为 1.3s。

2 结束语

本文介绍的容错系统为通过网络连接的 PC 双机备份服务器系统。系统采用监视器进程 Watchdog 定时检测工作节点状态，从而实现对单节点静默失效的容错和偶发失效容错。这种软件实现的分布式容错的主要优点是开发成本低，容错功能易于扩充，硬件系统易于升级。另一个优点是由于系统节点可以分布在不同的地理位置，可以实现对灾害性事件(例如地震和战争等)的容错。分布式容错系统的故障恢复时间比传统的硬件容错系统要长，这可以通过调整定时检测频度和提高网络带宽来加以改善。

作为一个实验系统，它所实现的对节点的控制流作定时检测以发现节点的静默失效并及时处理的功能，只是容错系统的基本功能的一部分。对该系统的容错功能的扩充和性能的完善的工作正在进行之中。

(收稿日期：1999－06－08)

(下转第 356 页，continued on page 356)

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.c

5 Operator's Manual of Lunar DPX System DXA's software: Technical Manual. Appendix F, (Version 3/95)
6 Hanson j. Standardization of femur BMD. J Bone Miner Res, 1997, 12(8): 1316

(收稿日期: 1998—10—05)

Comparative Study on Lunar, Hologic, Norland Dual Energy X-Ray Bone Densitometry with European Spine Phantom

Bao Ande[1] Ma Haibo[1] Qin Linlin[1] Chen Jinbiao[1] Zhang Yuanxun[2]
Li Jin[3] Sun Jihui[4] Yuan Yi[5] Ren Sumei[6] Sun Shihua[7] Yang Liping[8]

(1 Institute of Clinical Medical Sciences, China—Japan Friendship Hospital, Beijing, 100029; 2 Shanghai Institute of Nuclear Research, Academia Sinica; 3 Beijing Jishutan Hospital; 4 Beijing Chaoyang Hospital; 5 PLA General Hospital; 6 Beijing Youyi Hospital; 7 PLA General Logistic Clinic; 8 Beijing Tongren Hospital)

Abstract We had a European Spine Phantom (ESP) to be measured at three DXA machines each made by three corporations and six Lunar DPX system DXA machines so as to cross-calibrate measuring results. The results have good accordance in six Lunar DPX system machines; but the measuring results showed apparent differences among three corporations' DXA machines. Cross-calibration functions through our results have good accordance with the functions given by International DXA Standardization Committee(IDSC).

Key words: Dual energy X-ray bone densitonmetry(DXA) European Spine Phantom (ESP) Bordern determination Cross-calibration function

【作者简介】

包安德, 男, 31 岁, 研究实习员, 学士; 马海波, 男, 35 岁, 技师; 秦林林, 女, 55 岁, 研究员; 陈金标, 男, 30 岁, 研究实习员。

(上接第 351 页, continued from page 351)

A Fault Tolerant System by Using Distributed RTOS

Ge Yingan Liu Songqiang Wang Yanfang

(Fast Electronics Lab, University of Science and Technology of China, Hefei, 230027))

Abstract This paper describes the design and implementation of a prototypal distributed fault tolerant system, which is developed under QNX RTOS by networking two standard PCs. By using a watchdog timer for error detection, the system can be tolerant for fail silent and transient fault of a single node.

Key words: Distributed system Fault-tolerance Real time

【作者简介】

戈应安, 男, 博士生; 刘松强, 男, 研究员, 博导; 王砚方, 男, 61 岁, 教授, 博导。

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.c