

## Replicação de Objetos em um Sistema Distribuído

Emerson Rogério de Oliveira Junior<sup>1</sup> Philippe Olivier Alexandre Navaux<sup>2</sup>

Pós Graduação em Ciência da Computação  
Instituto de Informática – Universidade Federal do Rio Grande do Sul  
Av. Bento Gonçalves, 9500 – Bloco IV  
CEP 91501-970–Porto Alegre – RS – Brasil Tel. (051)3368399 (ramal 6168)  
email: emerson@inf.ufrgs.br

### RESUMO

Este trabalho apresenta a implementação de replicação de objetos em um sistema distribuído. O objetivo principal é avaliar o comportamento de uma aplicação em um sistema distribuído, mediante a ocorrência de uma falha de crash. O objeto que está sendo processado possui a função de coordenar uma ação, a qual está acontecendo em um grupo de objetos clientes.

A replicação permite que o objeto coordenador seja duplicado, sendo que outros objetos realizarão as mesmas funções do coordenador. É adotada a técnica do primário-backup onde, no caso de falha do objeto primário, um objeto réplica assume a função do objeto primário e continua a coordenar os objetos da aplicação.

**Palavras-Chave:** Sistemas distribuídos, objetos distribuídos, tolerância a falhas em sistemas distribuídos, replicação, DPC<sup>++</sup>, PVM.

### ABSTRACT

This work presents an implementation of object replication in a distributed system. The main goal is to verify the behavior of a distributed system when a fault occurs in an object that coordinates a determined action in a group of client objects.

The replication technique allow that the object coordinator will be duplicated. This duplicated object will execute the same functions of the initial coordinator. This work uses the primary-backup approach where, in case of primary object fault, one replica assumes the primary object function and continues controlling the objects of the application.

**Keywords:** Distributed systems, distributed objects, fault tolerance in distributed systems, replication, DPC<sup>++</sup>, PVM.

## 1 Introdução

Técnicas de tolerância a falhas são utilizadas em sistemas distribuídos visando detectar a ocorrência de erros produzidos por falhas e recuperá-los, para continuar uma operação e retornar à computação normal [8]. Uma das técnicas é conhecida como replicação, a qual permite que objetos possam ser duplicados

<sup>1</sup> Professor da Universidade de Passo Fundo - UPF e da Universidade Regional Integrada URI – Câmpus Frederico Westphalen (ambas no RS) e mestrando do CPGCC/UFRGS: Arquitetura de Computadores, Processamento Distribuído, Tolerância a Falhas em Sistemas Distribuídos. e-mail: emerson@inf.ufrgs.br

<sup>2</sup> Professor CPGCC/UFRGS: Dr. Eng. em Informática (Instituto Nacional Politécnico de Grenoble – França 1979): Arquitetura de Computadores, Processamento Paralelo, Avaliação de Desempenho. email: navaux@inf.ufrgs.br

fazendo com que, na ocorrência de falha em um objeto, uma cópia do objeto possa assumir as funções do objeto inicial.

Objetos participantes de uma transação devem ser implementados de forma que o processamento executado continue, mesmo na ocorrência de falhas [8]. Duas técnicas podem ser utilizadas para garantir a replicação de objetos: técnica de réplicas ativas e técnica do primário-backup [5].

A técnica do primário-backup utiliza um servidor primário e servidores backups do primário. Todos eles conectados por um link de comunicação [5] e [9]. Um cliente inicialmente envia uma mensagem para o primário. Quando o primário recebe a mensagem, ele a processará e atualizará o seu estado, se necessário. Depois envia a atualização para o seu backup.

Este artigo apresenta um modelo de replicação de objetos distribuídos no DPC<sup>++</sup> [6], baseado na técnica do primário-backup.

## 2 Trabalhos Correlatos

Poucos trabalhos que utilizam replicação de objetos distribuídos, utilizando-se a técnica do primário-backup, são encontrados na literatura. No que diz respeito a processos, aparecem as implementações de [7] e [1]. A aplicação do primário-backup em um sistema de arquivos está presente em [3]. Uma configuração de hardware extra a nível de processadores, sendo um primário e outro backup, é implementada nos computadores Tandem [2]. Os protocolos não-bloqueantes são o objeto do trabalho de [4].

## 3 Modelo da Replicação de Objetos Distribuídos

O modelo para o objeto Diretório no DPC<sup>++</sup> tem o objetivo principal de garantir a tolerância a falhas através da replicação deste objeto. A figura 3.1 apresenta o modelo da replicação de objetos distribuídos utilizado neste trabalho.

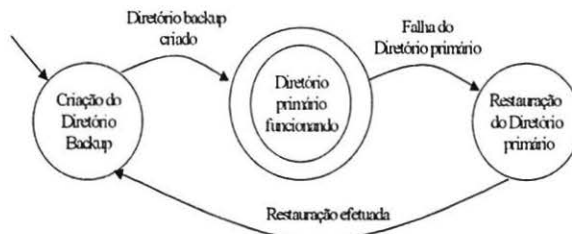


FIGURA 3.1—Autômato Definindo o Modelo de Replicação do Diretório.

## 4 Validação do Modelo

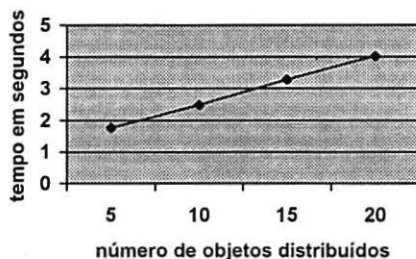
O modelo foi implementado em uma rede de PC's com 5 estações conectadas entre si. Foi utilizado Linux versão Debian 1.3.0 e a biblioteca PVM – (Parallel Virtual Machine) versão beta 3.4. O compilador C<sup>++</sup> utilizado foi o g<sup>++</sup>.

Cada processo controla a execução de um único objeto. A aplicação realiza troca de mensagens entre os objetos distribuídos e o objeto Diretório.

## X Simpósio Brasileiro de Arquitetura de Computadores

Foram realizados dois experimentos para verificar o comportamento da aplicação, em relação ao tempo de execução. O primeiro experimento levou em consideração a execução da aplicação utilizando-se objetos distribuídos, sem a ocorrência de falha no objeto Diretório.

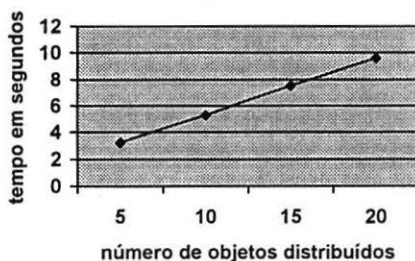
**GRÁFICO 4.1: Distribuição do Tempo de Execução sem a Ocorrência de Falha no Objeto Diretório**



Os valores apresentados no gráfico 4.1 representam o tempo médio encontrado quando utilizados 5, 10, 15 e 20 objetos distribuídos trocando mensagens com o objeto Diretório.

O segundo experimento apresenta a aplicação com a ocorrência de falha no objeto Diretório. O gráfico 4.2 mostra os valores médios de tempo encontrados quando se utiliza 5, 10, 15 e 20 objetos distribuídos. Nestes tempos se encontra computado o *overhead* causado para se conseguir a replicação do objeto Diretório.

**GRÁFICO 4.2: Distribuição do Tempo de Execução com a Ocorrência de Falha no Objeto Diretório**



## 6 Conclusões

Foi apresentado e validado um modelo de implementação utilizando a técnica de replicação conhecida como primário-backup, levando-se em consideração

## X Simpósio Brasileiro de Arquitetura de Computadores

o hardware e o software disponíveis. Este modelo garantiu que, mesmo no caso de falha do objeto Diretório, a execução da aplicação continuou até o seu final, conforme foi demonstrado nos gráficos 4.1 e 4.2.

Nos experimentos realizados, foi verificado que o custo extra de processamento e troca de mensagens decorrente da implementação do modelo da replicação do objeto Diretório ficou em torno de 1,4 s. Este valor será sempre o mesmo, quer o sistema possua 5, 10, 15, 20 ou mais objetos participantes, comunicando com o objeto Diretório. Estes cálculos vêm confirmar a correção do modelo proposto e implementado.

Conforme o modelo, independente do número de objetos participantes da transação, o custo extra de processamento deverá se manter o mesmo, no caso de falha do objeto Diretório.

Um dos grandes objetivos a ser atingido na execução de uma aplicação é a confiabilidade. Deve-se garantir que a aplicação do usuário continue executando, mesmo na ocorrência de falhas.

Este trabalho demonstrou que a confiabilidade é possível de ser alcançada mediante a utilização de técnicas de replicação de objetos.

### Referências Bibliográficas

- [1] ALSBERG, P. A. and DAY, J.D. **A Principle for Resilient Sharing of Distributed Resources**. Proc. of the Second Intern. Conf. on Software Engineering. San Francisco, CA. p.562-570, 1976.
- [2] BARTLETT, J. F. **A NonStop Kernel**. Proceedings of the Eighth Symposium on Operating Systems Principles. In ACM Operating Systems Review. V.15, n.5, 1981.
- [3] BHIDE, A. et al. **A Highly Available Network File Server**. Proc. of the USENIX. P.199-205, 1991.
- [4] BUDHIRAJA, N. and MARZULLO, K. **Tradeoffs in Implementating Primary-Backup Protocols**. Dept. of Computer Sc., Cornell Univ. Tech. Report TR 92-1307, Ithaca, Ny, 1992.
- [5] BUDHIRAJA, N. et al. **The Primary-Backup Approach**. In: Distributed Systems. ACM Press. New York, p.199-216, 1993.
- [6] CAVALHEIRO, G. et al. **DPC<sup>++</sup>: Uma Linguagem para Processamento Distribuído**. Florianópolis: V SBAC-PAD, anais..., outubro, 1993.
- [7] GHOSH, S. et al. **Fault-Tolerance Through Scheduling of Aperiodic Tasks in Hard Real-Time Multiprocessos Systems**. **IEEE Trans. on Parallel and Distributed Systems**. New York, v. 8, n. 3, p.272-284, mar. 1997.
- [8] JALOTE, P. **Fault Tolerance in Distributed Systems**. New Jersey: PTR Prentice Hall, Englewood Cliffs, 1994.
- [9] MULLENDER, S. **Distributed Systems**. Addison Wesley Publishing Company. ACM Press. New York. 1995.