COMPUTING LABORATORY

The Design of a
Reliable Remote
Procedure Call
Mechanism

by

~~S.K.~~ Shrivastava
and Fabio Panzieri

10/81   81 pages -

Abstract:

In this paper we describe the design of a reliable Remote Procedure Call mechanism intended for use in Local area networks. Starting from the hardware level that provides primitave facilities for data transmission, we describe how such a mechanism can be constructed. We discuss various design issues involved, including the choice of a message passing system over which the remote call mechanism is to be constructed and the treatment of various abnormal situations such as lost messages and node crashes. We also invest-igate what the reliability requirements of the Remote Procedure Call mechanism should be with respect to both the application programs using it and the message passing system on which it itself is based.

The Design of a Reliable Remote Procedure Call Mechanism

By

S.K. Shrivastava and F. Panzieri

TECHNICAL REPORT SERIES

bibliographical details

SHRIVASTAVA, Santosh Kumar

The design of a reliable Remote Procedure Call mechanism.
[By] S.K. Shrivastava, F. Panzieri.

Newcastle upon Tyne: University of Newcastle upon Tyne,
Computing Laboratory, 1981.

(University of Newcastle upon Tyne, Computing Laboratory,
Technical Report Series, no. 171.)

Added entries PANZIERI, Fabio
UNIVERSITY OF NEWCASTLE UPON TYNE.
Computing Laboratory. Technical Report Series. 171.

Suggested classmarks (primary classmark underlined)
Dewey (18th): 001.64404
U.D.C. 519.687

Suggested keywords

ATOMIC ACTIONS   FAULT TOLERANCE
DATA COMMUNICATIONS   LOCAL AREA NETWORKS
DISTRIBUTED SYSTEMS   RELIABILITY

Abstract

In this paper we describe the design of a reliable Remote Procedure Call mechanism intended for use in local area networks. Starting from the hardware level that provides primitive facilities for data transmission, we describe how such a mechanism can be constructed. We discuss various design issues involved, including the choice of a message passing system over which the remote call mechanism is to be constructed and the treatment of various abnormal situations such as lost messages and node crashes. We also investigate what the reliability requirements of the Remote Procedure Call mechanism should be with respect to both the application programs using it and the message passing system on which it itself is based.

About the author
Dr. Shrivastava joined the Computing Laboratory of the University of Newcastle upon Tyne in August 1975, where he is currently a Lecturer.

Mr. Panzieri joined the Computing Laboratory of the University of Newcastle upon Tyne in June 1979, where he is currently a Research Associate.

1. Introduction

In this paper we describe the design of a reliable Remote Procedure Call (RPC) mechanism which we have been investigating within the context of programming reliable distributed applications. In the following we consider a distributed system as composed of a number of interacting "client" and "server" processes running on possibly distinct nodes of the system; the interactions between a client and a server are made possible by the suitable use of the RPC mechanism. Essentially in this scheme, a client's remote call is transformed into an appropriate message to the named server who performs the requested work and sends the result back to the client and so terminating the call. The RPC mechanism is thus implemented on top of a message passing interface. Some of the interesting problems that need to be faced are: i) the selection of appropriate semantics and reliability features of the RPC mechanism, ii) the design of an appropriate message passing interface over which the RPC is to be implemented, and iii) the treatment of abnormal situations such as node crashes. These problems and their solutions are discussed in this paper. We shall concentrate primarily on the relevant reliability issues involved, so other directly or indirectly related issues such as type checking, authentication and naming will not be addressed here.

The RPC mechanism described in the following has been designed for a local area network composed of a number of PDP 11/45 and LSI 11/23 computers (nodes) interconnected by the Cambridge Ring[1]; each node runs the Unix* (V7) operating system. However, most of the ideas presented in this paper are, we believe, sufficiently general to be applicable to any other local area network system.

---

* Unix is a Trademark of Bell Laboratories.