

C
001-642
RAT

PERFORMANCE ANALYSIS OF CODA FILE SYSTEMS

A PROJECT REPORT PRESENTED BY

RATHNAYAKE MUDIYANSELAGE NEVIL BANDARA RATHNAYAKE

to the Board of Study in Statistics and Computer Science of the

POST GRADUATE INSTITUTE OF SCIENCE

*in partial fulfillment of the requirement
for the award of the degree of*

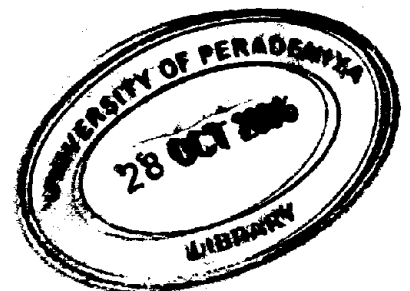
MASTER OF SCIENCE IN COMPUTER SCIENCE

of the

UNIVERSITY OF PERADENIYA

SRI LANKA

2005



603389

PERFORMANCE ANALYSIS OF CODA FILE SYSTEM

R. M. Nevil Bandara Rathnayake

Distributed file systems provide access to data stored at servers using file system interfaces. These systems are useful from the point of view of the users due to many reasons. Data sharing of multiple users, user mobility, location transparency, location independence and backups and centralized management are some of them.

Coda is one such distributed file system and it makes file available to a collection of client computer as part of their directory tree, but ultimately maintains the authoritative copy of the file data on servers. In addition to common facilities provided by distributed file system Coda has some more desirable features that makes it stand out among other distributed file systems.

In order to support disconnected operation, Coda provides full access to a caches section of the file space during voluntary or involuntary network or server outages, and Coda will automatically reintegrate the changes made on disconnected clients when reconnecting is one such feature.

Another important feature is the read write failover server replication where data is stored and fetch from any of a group of servers and Coda will continue to operate even only a subset of all servers is available.

The primary objective of this study was to implement Coda file system one a server running Windows NT 4 server operating system and a client machine running windows NT4

Workstation operating system. Such an implementation provides Unix like features on a windows platform.

The second objective was to analyze the performance of the Coda file system when the client cache manager was implemented as a user level process instead of having it implemented in kernel level.

The benchmark tests performed in the presence of the user level cache manager clearly showed improvement in the file system performance compared to when the cache manager was not present.

