High Service Reliability For Cluster Server System

M.Mat Deris, M. Rabiei, A. Noraziah, H.M. Suzuri

Department Of Computer Science
University College of Science and Technology
21030 Terengganu Malaysia

Agenda

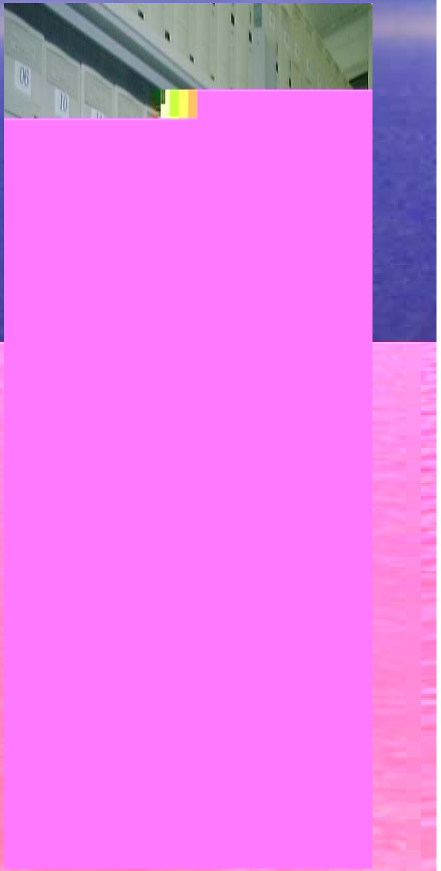
- Introductions
  - The Problems
  - The Solutions

- System Model
  - Neighbor Replica Distribution Technique (NRDT)
  - Cluster Server Architecture
  - Resource Distributed Agent

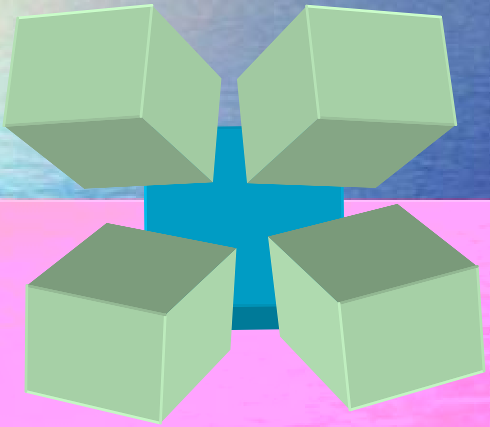
- Implementation
- Performance Comparison
- Conclusion

The Problems

- Reliability refers to the probability that the system under consideration does not experience any failures in a given time interval.
- A reliable cluster system (CS) is one that can continue to process user requests even when the underlying system is unreliable.
- Thus providing reliable and efficient services are the primary goals in designing a CS.



The Problems (cont)



- In order to provide reliable services, a CS needs to maintain the availability of the service types on some replicas.
- Therefore services-types replication plays an important role in the CS environment to become a highly reliable system

The Solutions

- Several techniques have been proposed in managing replicated data (in this case service-types). Different techniques have different reliability levels of managing replicated service-types.
- Since all of them replicate service-types to all servers in the cluster, storage capacity become an issue, thus an optimum number of servers to replicate the service-types is required.

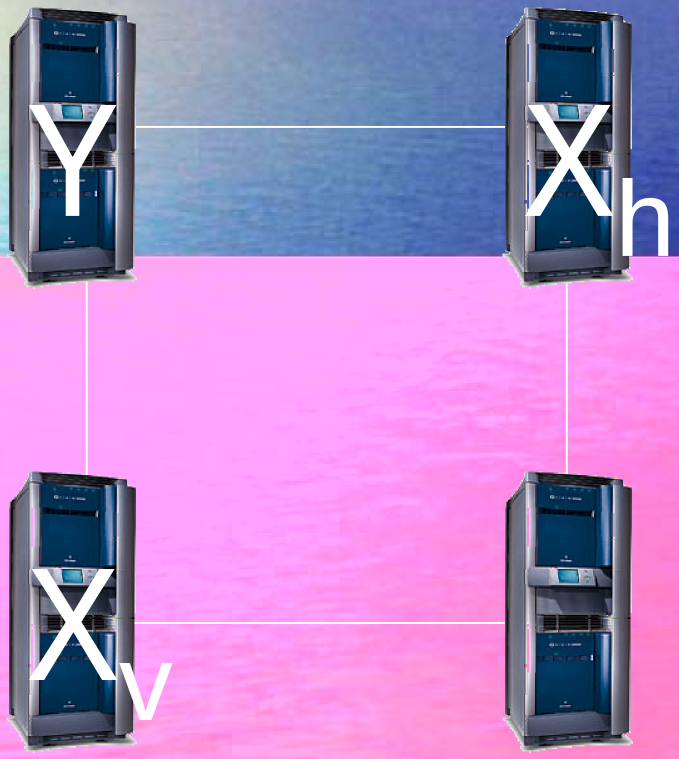
The Solutions (cont)

- Two-replica distribution technique (TRDT) is proposed by Shen, whereas, all service-types have two-replicas on different nodes and all nodes have two services-type replicas.
- The shortcoming is the system should have replica-availability more than 99% in order to achieve high reliability.

The Solutions (cont)

- We proposed neighbor-replica distribution technique (NRDT) to improve the reliability of the CS
- High availability is achieved by imposing a neighbor logical structure on service-types copy: a service-type from one node will be replicated to its neighboring nodes.

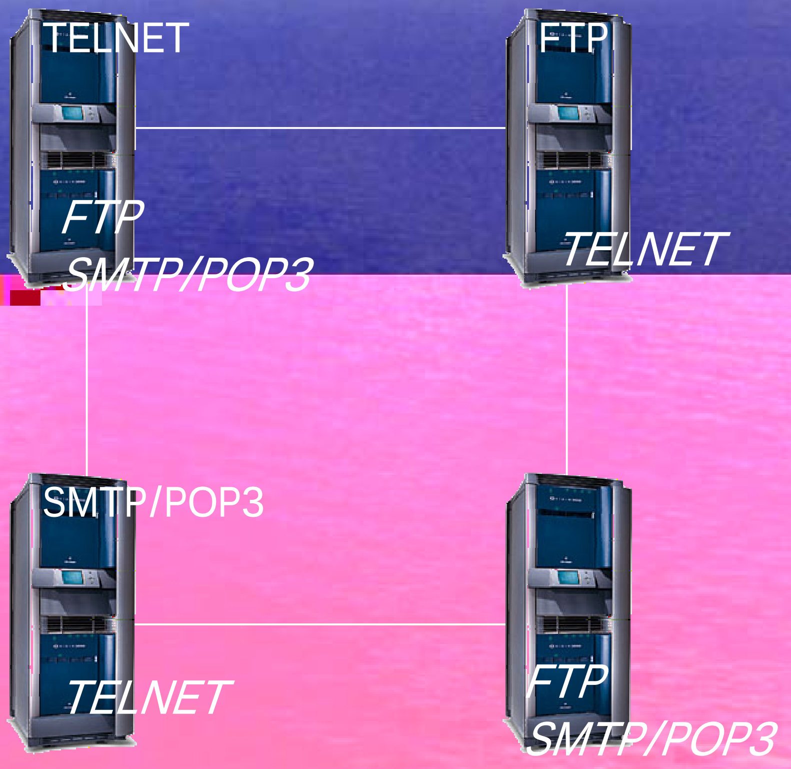
NRDT - Definitions



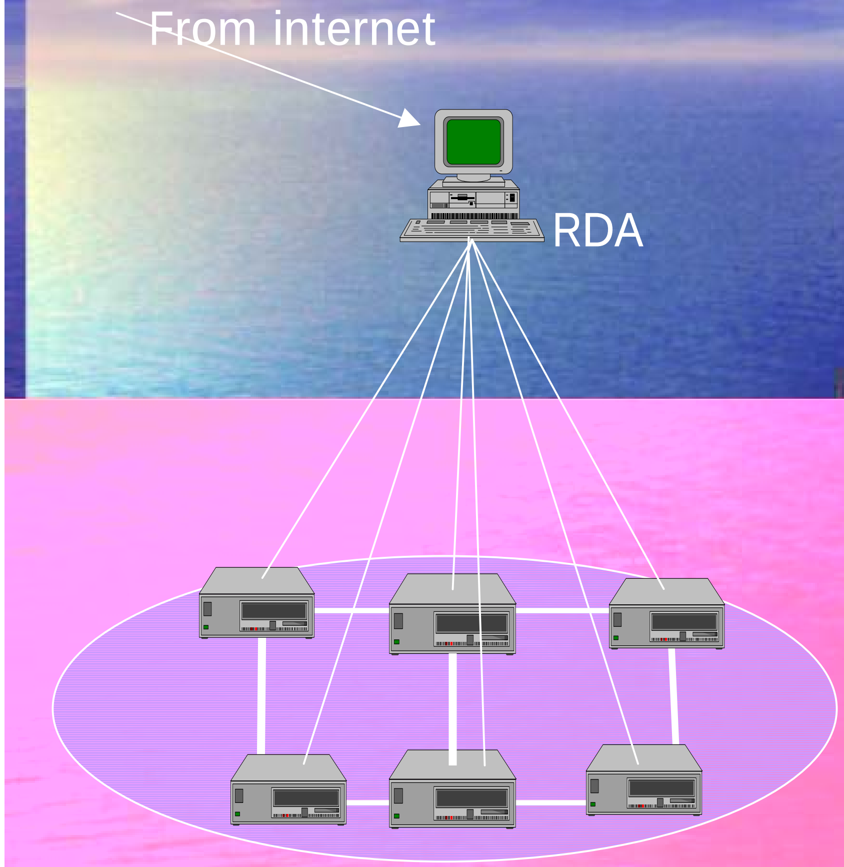
- NRDT is a cluster server technique that organizes the server node in the form of logical two-dimensional grid structure.
- A node X is a neighbor to Y, if X is logically-located vertical-horizontal to Y.

NRDT – Services & Replications

- Services
  - Each node has it own service
- Replications
  - Each node holds the service that belongs to its neighbor

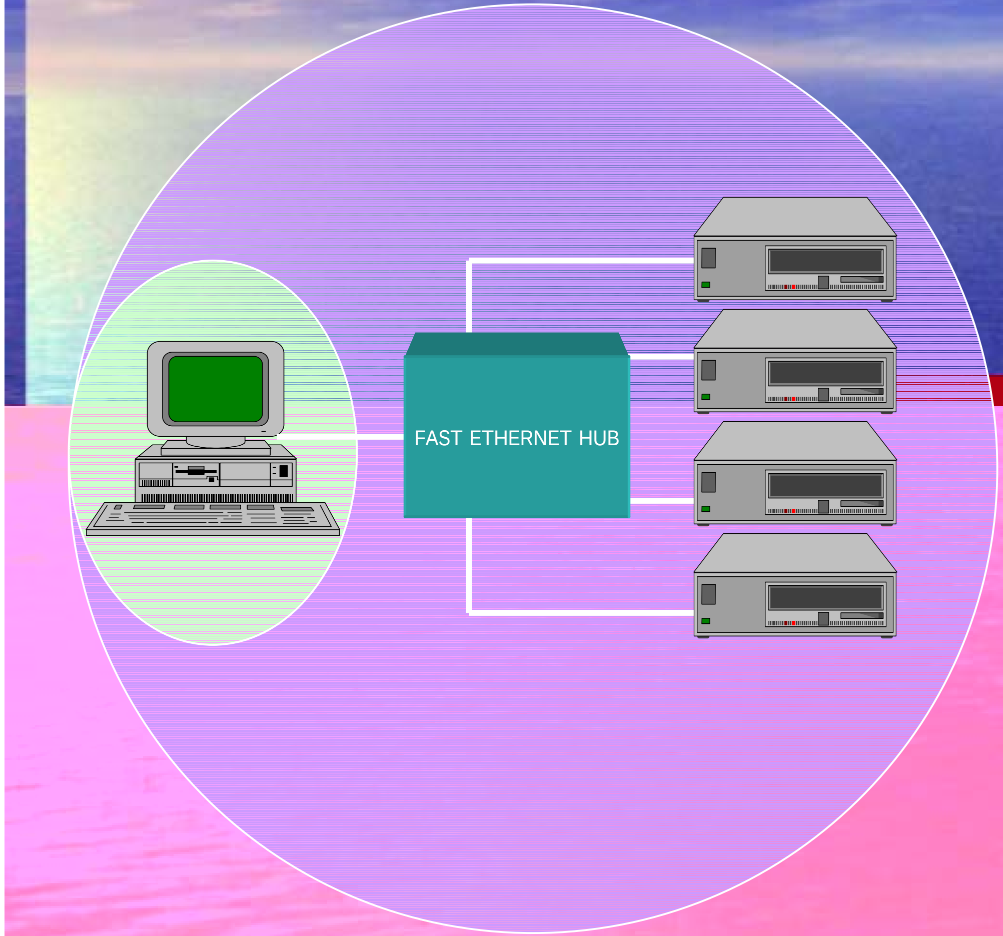


Cluster Server Architecture



- Consists of a RDA and a group of servers
- Each node supports:
  - Service-types
  - Replication Modules
    - Log Monitor
    - Propagator
    - Receiver
  - Network Interface

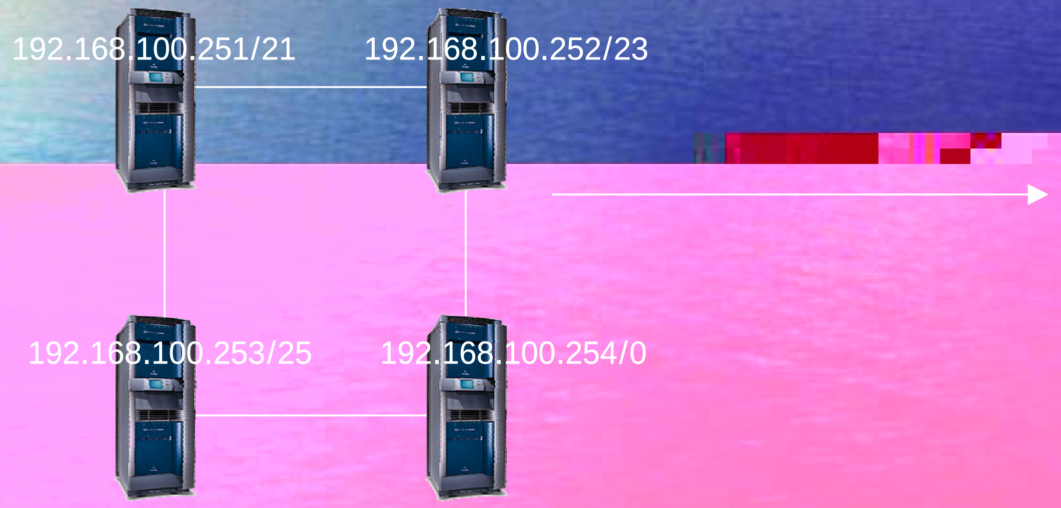
Resource Distributed Agent (RDA)



- Managing NRDT
- Managing Cluster Server

RDA – Managing NRDT

- Creating the “Neighborhood Environment”



(ipvalue.conf file)

192.168.100.251	21
192.168.100.252	23
192.168.100.253	25
192.168.30.254	0

(rda.conf file)

192.168.100.251	21	N	192.168.100.253	N	192.168.100.252
192.168.100.252	23	N	192.168.100.254	192.168.100.251	N
192.168.100.253	25	192.168.100.251	N	N	192.168.100.254

RDA – Managing Cluster Server

- Serve the requested services

```
telnet stream tcp nowait root /usr/sbin/tcpd rdagent telnet /etc/rda.conf in.telnetd
ftp stream tcp nowait root /usr/sbin/tcpd rdagent ftp /etc/rda.conf proftpd
pop3 stream tcp nowait root /usr/sbin/tcpd rdagent pop3 /etc/rda.conf /usr/sbin/pop3d
```

- Do recovery when primary host went down

192.168.100.251	21	N	192.168.100.253	N	192.168.100.252
192.168.100.252	23	N	192.168.100.254	192.168.100.251	N
192.168.100.253	25	192.168.100.251	N	N	192.168.100.254

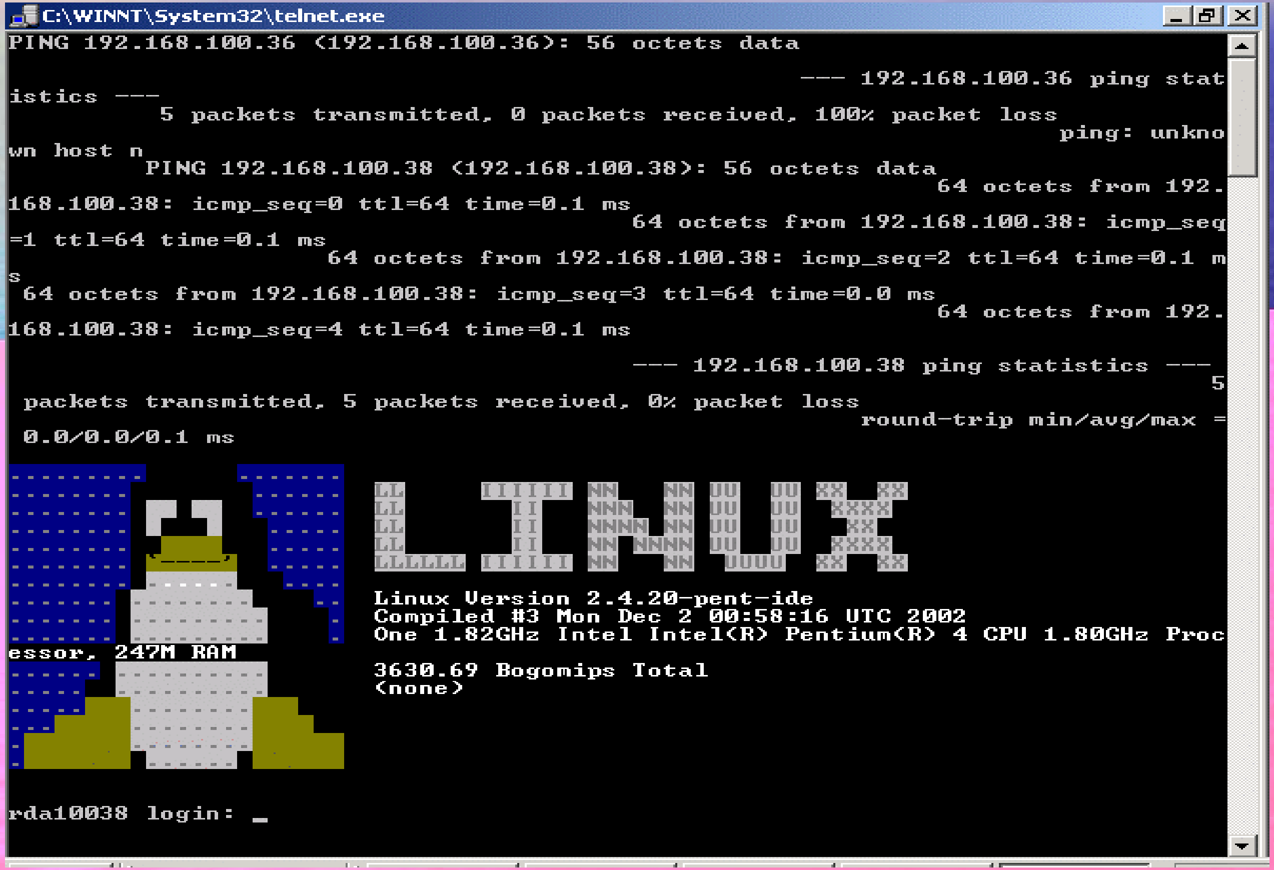
Implementation

```
WINSOCK.DLL: WinSock 2.0
WS_FTP Pro 6.05 2000.01.17, Copyright © 1992-2000 Ipswitch, Inc.
- -
connecting to 192.168.30.252:21
Connected to 192.168.30.252 port 21
PING 192.168.100.35 (192.168.100.35): 56 octets data
--- 192.168.100.35 ping statistics ---
5 packets transmitted, 0 packets received, 100% packet loss
ping: unknown host n
PING 192.168.100.37 (192.168.100.37): 56 octets data
64 octets from 192.168.100.37: icmp_seq=0 ttl=64 time=0.1 ms
64 octets from 192.168.100.37: icmp_seq=1 ttl=64 time=0.1 ms
64 octets from 192.168.100.37: icmp_seq=2 ttl=64 time=0.0 ms
64 octets from 192.168.100.37: icmp_seq=3 ttl=64 time=0.1 ms
64 octets from 192.168.100.37: icmp_seq=4 ttl=64 time=0.1 ms
--- 192.168.100.37 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.0/0.0/0.1 ms
220 ProFTPD 1.2.6 Server (ProFTPD Default Installation) [rda10037.lipur.org]
USER ahmad
331 Password required for ahmad.
PASS (hidden)
230 User ahmad logged in.
PWD
257 "/home/ahmad" is current directory.
SYST
215 UNIX Type: L8
Host type (S): UNIX (standard)
PORT 192,168,30,99,14,175
200 PORT command successful
LIST
150 Opening ASCII mode data connection for file list
Received 0 bytes in 0.1 secs, (0.00 bps), transfer succeeded
226 Transfer complete.
```

FTP Log

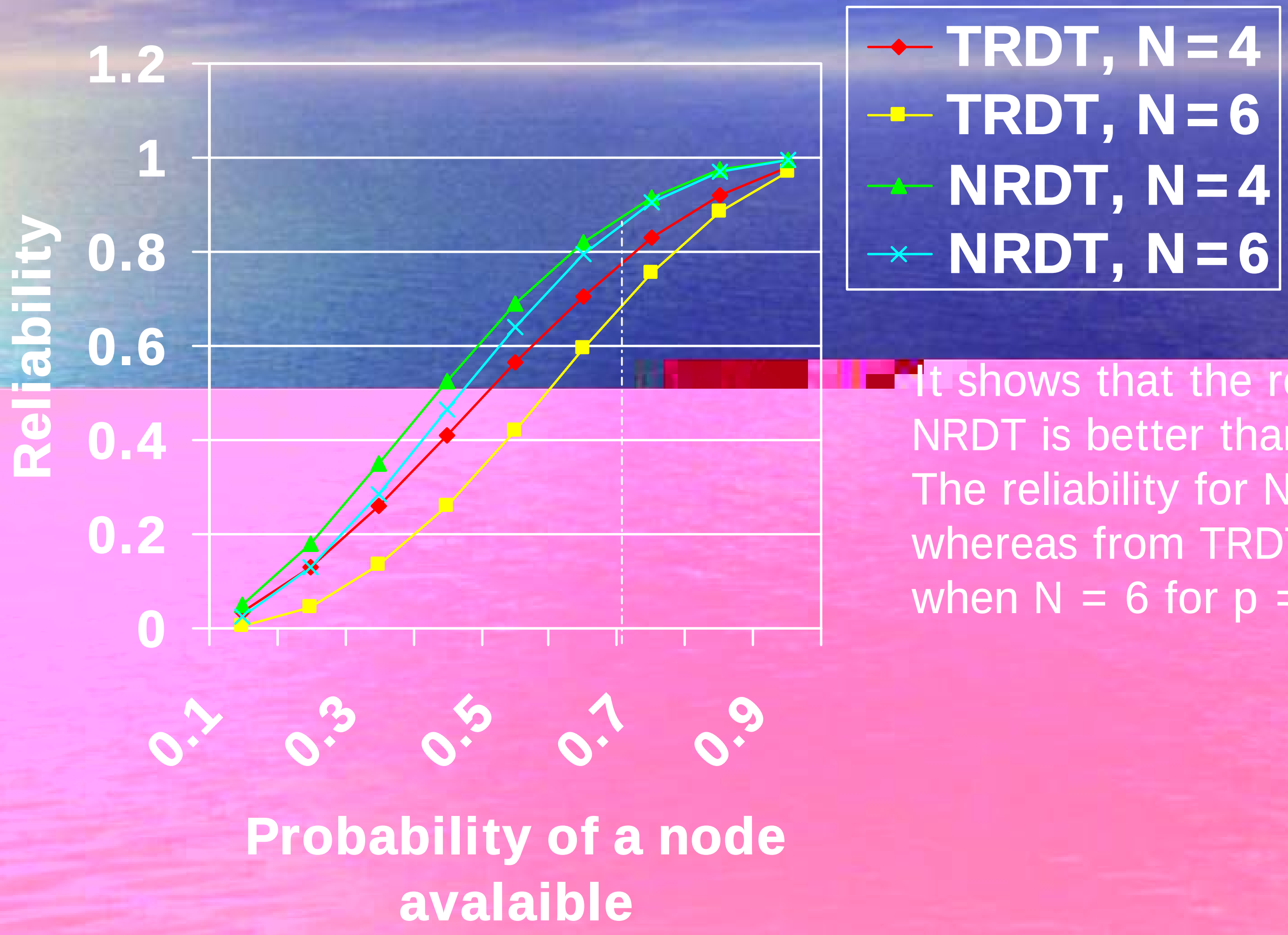
File

Implementation (cont)



```
PING 192.168.100.36 (192.168.100.36): 56 octets data
                                                                            --- 192.168.100.36 ping stat
istics ---
          5 packets transmitted, 0 packets received, 100% packet loss
                                                                     ping: unkno
wn host n
         PING 192.168.100.38 (192.168.100.38): 56 octets data
                                                             64 octets from 192.
168.100.38: icmp_seq=0 ttl=64 time=0.1 ms
                                         64 octets from 192.168.100.38: icmp_seq
=1 ttl=64 time=0.1 ms
                     64 octets from 192.168.100.38: icmp_seq=2 ttl=64 time=0.1 m
s
 64 octets from 192.168.100.38: icmp_seq=3 ttl=64 time=0.0 ms
                                                             64 octets from 192.
168.100.38: icmp_seq=4 ttl=64 time=0.1 ms

                                         --- 192.168.100.38 ping statistics ---
                                                                               5
 packets transmitted, 5 packets received, 0% packet loss
                                                         round-trip min/avg/max =
 0.0/0.0/0.1 ms

                        Linux Version 2.4.20-pent-ide
                        Compiled #3 Mon Dec 2 00:58:16 UTC 2002
                        One 1.82GHz Intel Intel(R) Pentium(R) 4 CPU 1.80GHz Proc
essor, 247M RAM

                        3630.69 Bogomips Total
                        <none>


rda10038 login: _
```

Performance Comparison



It shows that the reliability under NRDT is better than TRDT. Example The reliability for NRDT is ~90% whereas from TRDT it is ~75% when N = 6 for p =0.7.

Conclusion

- RDA
  - It was developed and implemented successfully.
- NRDT
  - With NRDT technique, it provides a convenience approach to high reliability in CS system