ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 8, August 2013

Robust and Scalable Structure for Server-Interconnections in Data Centers

K.Chaitanya[1], Purushotham.P[2], Dr.R.V.Krishnaiah[3]

M.Tech Student,Research Scholar, Dept. of CS, DRK Institute of Science & Technology, Hyderabad, AP India[1]

Associate Professor, Dept. of CSE, DRK Institute of Science & Technology, Hyderabad, AP, India[2]

Principal, Department of CSE, DRK Group of Institutions, Hyderabad, Andhra Pradesh, India[3]

Abstract: Data center networking is the result of many server machines interconnected. The present data centers are connected using network switches organized in the form of a tree hierarchy. However, they are unable to meet the goals such as establishing data center network with low cost equipment, fault tolerance and balancing the capacity of network. Recently Dan Li et al. presented a new server-interconnection structure. This structure makes use of the both Ethernet ports that come with commodity server for server-interconnections. Thus it makes the data center structure highly scalable and efficient. In this paper we implement this structure by implementing a custom Java simulator. We also implement traffic aware routing mechanisms to improve robustness of the data center. The experimental results revealed that the proposed design is effective and robust to failures.

Index Terms –Data center, server interconnection structure, scalable routing, and fault tolerance

I.INTRODUCTION

Data center networking is made up of multiple servers and supported switches and other equipment. The server-interconnection is achieved using a structure and related protocols [1]. Thousands of servers are interconnected to form a data center [2], [3], [4]. The goal of data centers is to have such inter-connections with low cost equipment, robust to failures. The data center is supposed to provide infrastructure services such as Dryad [5], Map-reduce [6] and GFS [7]. As explored in [1] and [8] it is understood that the present data center networks are built over tree based hierarchical structure that fails to meet the requirements of a data center.

Dan Li et al. [9] studied the problem of a structure of data center which is scalable, low cost, and robust to failures. They have studied the present data center networks maintained by Yahoo, Google and Microsoft which are cash-rich companies. They proposed a new structure for data centers which can be managed with low cost equipment besides making it robust to failures. They built a structure which achieves server-interconnections by using both Ethernet ports that are shipped with the commodity servers. Thousands of servers can be inter-connected using both the port numbers. Thus it is possible to avoid costly switches required by a tree-hierarchy based network structure. They also considered to novel aspects in routing. They are balancing routes and traffic aware routing. The computation of traffic aware route makes the network structure more robust and highly scalable as it can make use of bandwidth optimally.

In this paper we implemented the network structure proposed by Dan Li et al. [9]through a custom Java simulator. The simulator demonstrates the proof of concept of a network structure for a real world data center. Simulation results reveal that the structure is robust to failures and it is highly scalable and effective. The remainder of this paper is structured as follows. Section II reviews relevant literature. Section III provides information about the proposed data center network architecture. Section IV presents the experimental results while section V concludes the paper.

II. PRIOR WORK

In this section we present three different structures that came into existence for server-interconnection in datacenters. The two recent proposals include D Cell [8] and Fat-Tree [1]. At present data center structure is made up of servers and switches in tree hierarchy based network. The switches are very costly in nature. The commodity servers come with two Ethernet ports. These ports were used for different purposes instead using the both for server-interconnections to achieve

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 8, August 2013

low cost solution for data centers. The high cost switches also cause single point of failures.

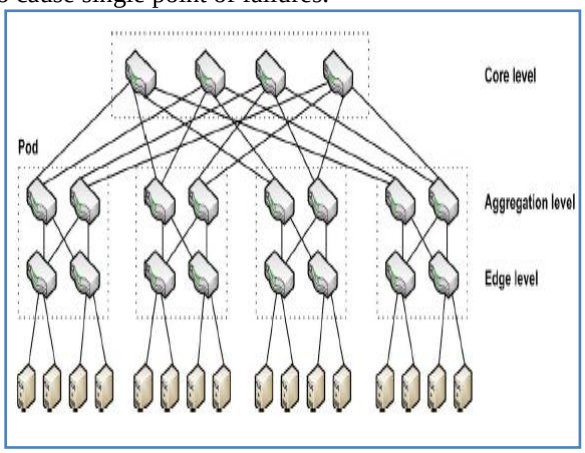

Fig. 1 –A Fat-Tree structure with three levels of switches

As can be seen in fig. 1, the structure has three levels such as core level, aggregation level and edge level. The problem with this structure is only one port of Ethernet in commodity server is used for server-interconnection. Another problem is that the server – interconnection intelligence is associated with switches making this structure costly.

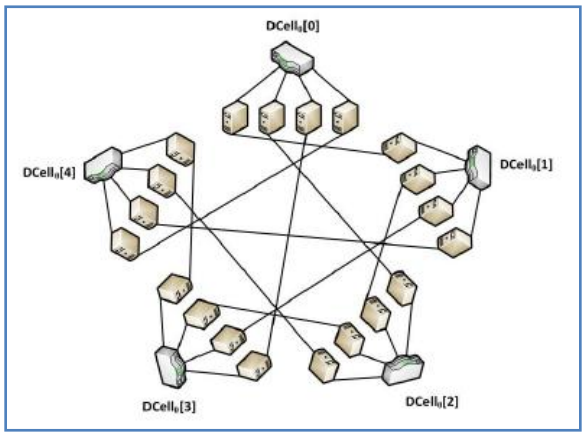

Fig. 2- DCell structure with 5 DCells

As can be seen in fig. 2, the DCell model structure for data center is presented. Unlike the other structure, this keeps the server-interconnection intelligence in server nodes only. However, it also has some drawbacks. One important drawback is that the network uses only one Ethernet port out of two provided by commodity servers. Thus it can't provide more connections.

With respect to other areas (not data centers) the interconnection is achieved using various techniques such as Dragonfly [10], Flattened Butterfly [11], De Bruijin [12], Torus [13], Butterfly [14], Hypercube [15], [16], and Ring [13]. Besides these structures other structures include parallel computing [17], [14], [13], switching fabric [18] and on-chip network [19]. Out of all these structures only Ring is something similar to the structure proposed in [9] where a novel structure is proposed for real world data center. The structure makes use of server-interconnections that make use of two ports provided by commodity servers.

III.PROPOSED INTERCONNECTION STRUCTURE

The server-interconnection structure proposed by Dan Li et al. [9] is described here. It is a structure which does not rely on costly switches. This is because it focuses on the server nodes for interconnection intelligence instead of switches. Thus it can reduce the cost or the establishment of data center. Moreover, every server comes with two Ethernet ports in the data center. The current structures use only one Ethernet port on each commodity server for interconnection purposes while the other port is used for backup. However, the approach described here makes use of both the Ethernet ports for server-interconnections only. This will make it easy to have thousands of servers to be connected with ease. It also implements the mechanisms for traffic aware routing. This will make this structure highly cost effective, scalable and robust to link failures. More details about this novel structure can be found in [9]. The algorithms used are presented here.

```
01 FiConnConstruct(k){
02   for(i1 = 0; i1 < gk; i1++)
03     for(j1 = i1 * 2^k + 2^(k-1) - 1; j1 < N_(k-1); j1 = j1 + 2^k)
04       i2 = (j1 - 2^(k-1) + 1)/2^k + 1
05       j2 = i1 * 2^k + 2^(k-1) - 1
06       connect servers [i1, j1] with [i2, j2]
07   return
08 }
```

Fig. 3 –Algorithm to construct structure

As can be seen in fig. 3, the server-interconnection is made programmatically to simulate a real world data center structure.

```
/*s: current server.
dst: destination server of the packet to be routed.
*/
01 TORoute(s, dst){
02   l = lowestCommonLevel(s, dst)
03   if(l == 0)
04     return dst
05   (i1, i2) = getLink(s, dst, l)
06   if(i1 == s)
07     return i2
08   return TORoute(s, i1)
09 }
```

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 8, August 2013

Fig. 4–Algorithm for traffic oblivious routing

As can be seen in fig. 4, the algorithm is meant for routing in the new structure built for data centers. This routing is normal and does not support traffic aware routing.

```
01 TARoute(s, pkt){
02   if(pkt.dst == s) /*This the destination*/
03     return NULL /*Deliver pkt to upper layer*/
04   if(pkt.phop == RTable[pkt.flow].nexthop) /*SRR*/
05     nhop = RTable[pkt.flow].prevhop
06     RTable[pkt.flow] = NULL
07     if(nhop ≠ NULL) /*This is not source server*/
08       return nhop
09   if(s == pkt.pr[l]) /*This is the proxy server*/
10     pkt.pr[l] = BYONE
11   ldst = getPRDest(pkt) /*Check PR for proxy server*/
12   nhop = TORoute(s, ldst)
13   if(s == pkt.src and nhop ≠ hn and hb > zb)
14     nhop = hn
15   if(pkt.phop == hn and nhop ≠ hn)
       or (pkt.phop ≠ hn and hb ≥ zb)
16     resetPR(pkt.pr, l)
17     RTable[pkt.flow] = (pkt.phop, nhop)
18     if(nhop ≠ hn and vfc > 0)
19       vfc = vfc – 1 /*VF*/
20     return nhop
21   fwdhop = nhop
22   while(fwdhop == nhop)
23     fwdhop = bypassLink(s, pkt, l)/*Try to bypass*/
24   if(fwdhop == NULL) /*Cannot find a bypassing path*/
25     resetPR(pkt.pr, l)
26     RTable[pkt.flow] = (pkt.phop, nhop)
27     return nhop
28   vfc = vfc + 1 /*VF*/
29   return pkt.phop /*Proxy found, SRR*/
30 }
```

Fig. 5 –Algorithm for Traffic – Aware Routing

As can be seen in fig. 5, the algorithm is meant for traffic aware routing in the structure built in data center. The structure is made up of server-interconnections that make use of both Ethernet ports of the commodity servers.

IV. RESULTS

We built a prototype application which is a custom Java simulator for demonstrating the proof of concept of the new structure for server-interconnections in data center. The environment used for the experiments is a PC with 8 GB RAM, Core 2 Dual processor running Windows 7 operating system. Net Beans IDE is used to develop the simulator programs.

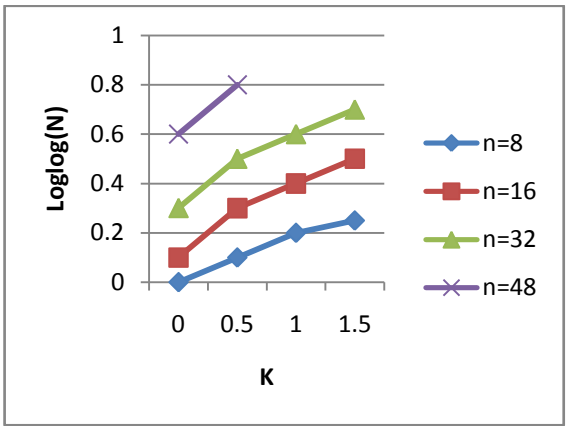

Fig. 6 –Relationship between number of servers and K

As can be seen in fig. 6, the results show the relationship between the number of servers and the structure level. As levels increase with number of servers showing linear relationship.

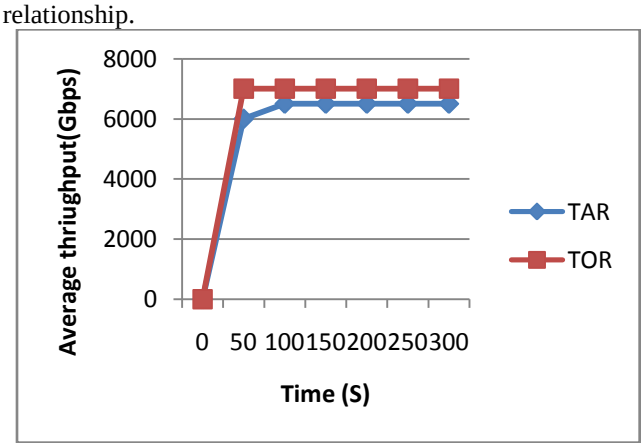

Fig. 7 –Aggregate throughput for random traffic

As can be seen in fig. 7, the horizontal axis shows time in seconds while the vertical axis represents average throughput. The results reveal the aggregate throughput for both TAR and TOR random traffic.

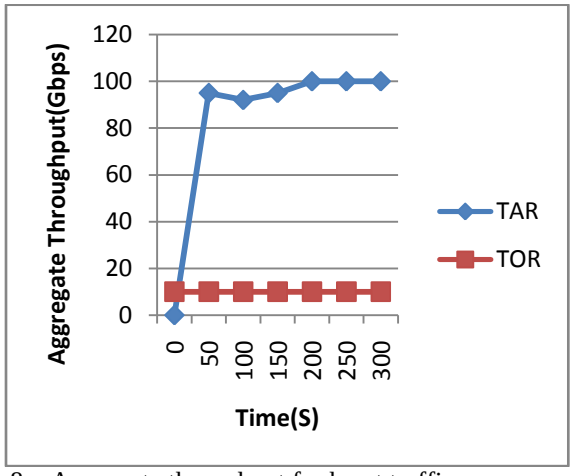

Fig. 8 – Aggregate throughput for burst traffic

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 8, August 2013

As can be seen in fig. 8, the horizontal axis shows time in seconds while the vertical axis represents aggregate throughput. The results reveal the aggregate throughput for both TAR and TOR burst traffic.

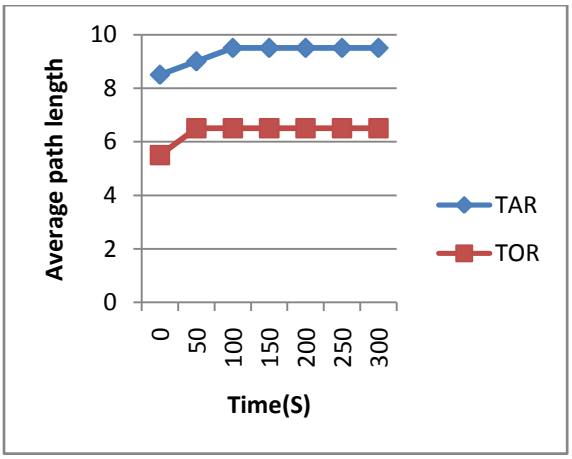

Fig. 9 – Average path length for burst traffic

As can be seen in fig. 9, the horizontal axis shows time in seconds while the vertical axis represents average path length. The results reveal the aggregate throughput for both TAR and TOR burst traffic.

V.CONCLUSION

In this paper we implement a structure for data center network which enables server-interconnection with low cost equipment, highly scalable and robust to failures. The structure implemented here is proposed by Dan Li et al.[9]. The server-interconnection is achieved by using the dual ports that come with commodity servers. Both ports are used for server interconnections. We implemented traffic aware routing mechanisms to make the structure robust to failures. We built a prototype in the form of custom Java simulator which demonstrates the proof of concept. The experimental results revealed that the new structure is scalable, efficient and robust to failures.

REFERENCES

[1] M. Al-Fares, A. Loukissas and A. Vahdat, "A Scalable, Commodity DataCenter Network Architecture", In Proceedings of ACM SIGCOMM'08,Aug 2008
[2] A. Carter, "Do It Green: Media Interview with Michael Manos",http://edge.technet.com/Media/Doing-IT-Green, Dec 2007
[3] L. Rabbe, "Powering the Yahoo! network", http://yodel.yahoo.com/2006/11/27/powering-the-yahoo-network, Nov 2006.
[4] T. Hoff, "Google Architecture", http://highscalability.com/googlearchitecture,Jul 2007
[5] M. Isard, M. Budiu, Y. Yu and etc., "Dryad: Distributed Data-ParallelPrograms from Sequential Building Blocks", In Proceedings of ACMEuroSys'07, 2007.
[6] J. Dean and S. Ghemawat, "MapReduce: Simplified Data Processing onLarge Clusters", In Proceedings of OSDI'04, 2004
[7] S. Ghemawat, H. Gobio, and S. Leungm, "The Google File System", InProceedings of ACM SOSP'03, 2003
[8] C. Guo, H. Wu, K. Tan, L. Shi, Y. Zhang and Songwu Lu, "DCell:A Scalable and Fault-Tolerant Network Structure for Data Centers", InProceedings of ACM SIGCOMM'08, Aug 2008
[9] Dan Li, Chuanxiong Guo, Haitao Wu, Kun Tan, Yongguang Zhang and Songwu Lu, "FiConn: Using Backup Port for Server Interconnection in Data Centers".
[10] J Kim, W. Dally, S. Scott and D. Abts, "Technology-Driven, Highly-Scalable Dragonfly Topology", In Proceedings of ISCA'08, Jun 2008.
[11] J Kim, W. Dally and D. Abts, "Flattened Butterfly: A Cost-EfficientTopology for High-Radix Networks", In Proceedings of ISCA'07, Jun2007.
[12] D. Loguinov, A. Kumar, V. Rai and S. Ganesh, "Graph-TheoreticAnalysis of Structured Peer-to-Peer Systems: Routing Distances and FaultResilience", In Proceedings of ACM SIGCOMM'03, Aug 2003
[13] B. Parhami, "Introduction to Parallel Processing: Algorithms and Architectures",Kluwer Academic, 2002
[14] F. Leighton, "Introduction to Parallel Algorithms and Architectures: Arrays.Trees.Hypercubes", Morgan Kaufmann, 1992
[15] H. Sullivan and T. R. Bashkow, "A large scale, homogeneous, fullydistributed parallel machine I", In Proceediings of ISCA'77, Mar 1977
[16] L. Bhuyan and D. Agrawal, "Generalized Hypercube and HyperbusStructures for a Computer Network", In IEEE TRANSACTIONS ONCOMPUTERS, 33(4):323-333, Apr 1984
[17] L. Bhuyan and D. Agrawal, "A general class of processor interconnectionstrategies", In Proceedings of ISCA'82, Apr 1982
[18] W. Dally, P. Carvey and L. Dennison, "The Avici Terabit Switch/Router",In Proceedings of Hot Interconnects'98, Aug 1998
[19] W. Dally and B. Towles, "Route Packets, Not Wires: On-Chip InterconnectionNetworks", In Proceedings of DAC'01, Jun 2001

BIOGRAPHY

K Chaitanya has completed B.Tech (C.S.E) from Auroras' Scientific & Technological Institute and pursuing M.Tech (C.S) in DRK - Institute of Science & Technology, JNTUH, Hyderabad, and Andhra Pradesh, India. Her main research interest includes Scalable and Cost-Effective Interconnection in Data Center Servers Using Dual Server Ports

Purushotham P is working as an Assistant Professor in DRK Institute of Science and Technology, JNTUH, Hyderabad,Andhra Pradesh,India. He is completed M.Tech in CSE from JNTUH, Hyderabad,Andhra Pradesh,India. His main research interest includes Data Mining, Compiler Design & Distributed Databases.

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 8, August 2013

Dr.R.V.Krishnaiah, did M.Tech (EIE) from NIT Waranagal, MTech (CSE) form JNTU, ,Ph.D, from JNTU Ananthapur, He has memberships in professional bodies MIE, MIETE, MISTE. His main research interests include Image Processing, Security systems, Sensors, Intelligent Systems, Computer networks, Data mining, Software Engineering, network protection and security control. He has published many papers and Editorial Member and Reviewer for some national and international journals.