A Failure Detection Model for Data Center Networks*

Kai Shen†
School of Software
Shanghai Jiao Tong University
Wallamaloo, New Zealand
knshen@sjtu.edu.cn

Charles Palmer
Palmer Research Laboratories
8600 Datapoint Drive
San Antonio, Texas 78229
cpalmer@prl.com

ABSTRACT

This paper provides a sample of a LaTeX document which conforms, somewhat loosely, to the formatting guidelines for ACM SIG Proceedings. It is an *alternate* style which produces a *tighter-looking* paper and was designed in response to concerns expressed, by authors, over page-budgets. It complements the document *Author's (Alternate) Guide to Preparing ACM SIG Proceedings Using LaTeX2$_\epsilon$ and BibTeX*. This source file has been written with the intention of being compiled under LaTeX2$_\epsilon$ and BibTeX.

The developers have tried to include every imaginable sort of "bells and whistles", such as a subtitle, footnotes on title, subtitle and authors, as well as in the text, and every optional component (e.g. Acknowledgments, Additional Authors, Appendices), not to mention examples of equations, theorems, tables and figures.

To make best use of this sample document, run it through LaTeX and BibTeX, and compare this source code with the printed output produced by the dvi file. A compiled PDF version is available on the web page to help you with the 'look and feel'.

CCS Concepts

•**Computer systems organization** → **Embedded systems;** *Redundancy;* Robotics; •**Networks** → Network reliability;

Keywords

ACM proceedings; LaTeX; text tagging

1. INTRODUCTION

Data center. The rest of this paper is organized as follows. Section 2 discusses related work,

*(Produces the permission block, and copyright information). For use with SIG-ALTERNATE.CLS. Supported by ACM.

†Dr. Trovato insisted his name be first.

WOODSTOCK '97 El Paso, Texas USA

DOI: 10.475/123_4

2. RELATED WORK

In this section, we will present various relevant works including failure detectors, failure detection architecture in distributed systems and failure localization in data center networks.

2.1 Failure Detectors

A failure detector(FD) is widely recognized as an oracle to intelligently suspect failed processes [2]. The monitored process periodically send heartbeat messages to its detector to prove its liveness.

Chen et al. proposed a FD that provides QoS [3]. It estimates the expected arrival time(EAs) of the next heartbeat message according to a slide window storing n most recent arrived messages. EAs determine the deadline that the detector will wait for the next heartbeat before suspecting the monitored process. *Chen FD* can adjust current network condition and set the timeout threshold adaptively by referring to recent heartbeats. Bertier et al. proposed a similar FD [1], whereas it uses a dynamic way to compute error margin of *Chen FD*. Other FDs like ϕ *Accrual FD* [5] and *Satzger FD* [6] can output an accrual failure value at any time which stands for the failure probability of the process.

2.2 Failure Detection Architecture

A failure detection architecture aims to provide the service of monitoring nodes in large scale distributed systems in a scalable way. Roughly speaking, there are two kinds of architectures. The first one is hierarchical architecture [4], all the nodes are partitioned into different groups, each group has a leader node. Within a group, the leader node is responsible for monitoring all the nodes. Leader nodes periodically send node status information of his group to other leader nodes. Hierarchical architecture can reduce the number of heartbeat messages effectively, but it exists *Single Point of Failure* problem, when the leader node crashes, it needs to select a new leader. Another alternative solution is Gossip-Style architecture [7]. Each node maintains a list containing the *heartbeat counter* for all the nodes in the system. Every T_{gossip} seconds, each node firstly update *heartbeat counter* of itself and then randomly select another node to send its list to. Upon receiving a message, the node will merge two lists and update each node's *heartbeat counter* with the bigger one. If the *heartbeat counter* does not update after T_{fail} seconds, the node will be marked as crashed. If the *heartbeat counter* does not update after $\mathrm{T}_{cleanup}$($\mathrm{T}_{cleanup} \geq \mathrm{T}_{fail}$) seconds, it will be removed from the list.

3. EVALUATION

4. CONCLUSIONS

This paper proposes a failure detector model for data center networks.

5. ACKNOWLEDGMENTS

This work is partially supported by the National High-Technology Research and Development Program of China under Grant No.2014AA01A301.

6. ADDITIONAL AUTHORS

7. REFERENCES

[1] M. Bertier, O. Marin, and P. Sens. Implementation and performance evaluation of an adaptable failure detector. In *Dependable Systems and Networks, 2002. DSN 2002. Proceedings. International Conference on*, pages 354–363. IEEE, 2002.

[2] T. D. Chandra and S. Toueg. Unreliable failure detectors for reliable distributed systems. *Journal of the ACM (JACM)*, 43(2):225–267, 1996.

[3] W. Chen, S. Toueg, and M. K. Aguilera. On the quality of service of failure detectors. *IEEE Transactions on computers*, 51(5):561–580, 2002.

[4] P. Felber, X. Défago, R. Guerraoui, and P. Oser. Failure detectors as first class objects. In *Distributed Objects and Applications, 1999. Proceedings of the International Symposium on*, pages 132–141. IEEE, 1999.

[5] N. Hayashibara, X. Defago, R. Yared, and T. Katayama. The φ accrual failure detector. In *Reliable Distributed Systems, 2004. Proceedings of the 23rd IEEE International Symposium on*, pages 66–78. IEEE, 2004.

[6] B. Satzger, A. Pietzowski, W. Trumler, and T. Ungerer. A new adaptive accrual failure detector for dependable distributed systems. In *Proceedings of the 2007 ACM symposium on Applied computing*, pages 551–555. ACM, 2007.

[7] R. Van Renesse, Y. Minsky, and M. Hayden. A gossip-style failure detection service. In *Middlewareąŕ98*, pages 55–70. Springer, 1998.