Implementation and performance evaluation of a real-time tele-monitoring system based on a private cloud platform

Yuan-Jen Chang[1,2,*], *Member, IEEE,* Chin-Hsing Chen[1], Wen-Tzeng Huang[3], Li-Feng Lin[4] , Chen-Yi Lee[5]

Presenter Disclosures

- **Presenter: Yuan-Jen Chang**
- **No relationships to disclose.**

Outline

- Introduction
- Objectives
- System architecture for real-time tele-monitoring
- Network architecture
- Proposed system architecture
- Testing results
- Discussions
- Conclusions

Introduction

- Heart disease ranks the second in the leading causes of death for Taiwanese people since 2007.
- Real-time monitoring an out-of-hospital cardiac patient's vital signs can reduce the risk of death.
- However, the high data transmission rate requirement for information such as electrocardiogram (ECG) needs to be addressed to ensure complete data transmission and storage in remote database servers.
- The challenge is how to provide high service availability while many users are accessing the database.

Objectives

- How could the tele-monitoring service benefit from cloud computing?
- Possibility to achieve the real-time ECG tele-monitoring through Internet?
- To evaluate the performance of private cloud under multi-users operation.

System architecture for real-time tele-monitoring

- A Web-based system was designed to provide the following functionalities:
  - (1) administrative management functions;
  - (2) tele-monitoring for single-lead ECG; and
  - (3) emergency alert and response.

Network architecture

- PART I indicates data communication between the database and the administrative member, while PART II indicates data communication between the database and the cadiac patients.
- WSN is used here as an example for demonstration. When a patient turns the sensor on, the sensor automatically registers the sensor status in the sensor profile built in the database.

The bottle neck

- Increasing the sampling rate will increase the latency of response.

How to conquer the difficulty

- Grid computing or cloud computing?
  - Grid computing is for application that requires high computing power
  - Cloud computing is for application that requires high database access
- The bottle neck is database for this scenario.
- To distribute the access loading of database to more server.

Implementation of private cloud

- Private cloud is implemented based on:
  - Windows Azure Platform (UPHI)
    - Windows Server 2008
    - MS SQL Serevr
    - C#
  - Citrix XenServer
    - CentOS
    - Apache+MySQL+PHP

- Java

The expression of vital sign data row

- In order to store vital signs data to database, some expressions are defined as follows:
  - Sampling rate: 512 Hz
  - 12bits for each sample
  - Every 6 bits data prefixes with “01” and converts to ASCII code
  - For example:

Clients operation simulation

- The experiment used Jmeter and Sikuli for automatic testing.
- Jmeter was used for multi-threads data transmission.
- Sikuli was used for administrative member actions

Testing results for less than 100 client users

- When clients users are less than 100, it shows the latency is kept in the same level for XenServer and Azure platform(UPHI).
- But the latency of XenServer is higher than that of Azure platform (UPHI).
- We show the results of Google App Engine (GAE) for comparison.

Testing results for 100 ~ 1000 client users

- The latency of the Azure platform (UPHI) was initially smaller

than that of the XenServer but increased gradually with the increase in client users.

- The latency of the Azure platform (UPHI) was greater than that of XenServer when the client users were greater than 250.
- The latency of XenServer keep between 2~10 seconds from 100 to 1000 client users.

Database loading and packages loss

- For Windows Azure platform (UPHI):
  - When client users exceed about 250, it appears packages loss in data transmission.
  - In addition, database server shows high access loading.
- For Citrix XenServer:
  - The data transmission is almost stable.

Discussions

- The latency for private clouds such as Azure and XenServer are kept in the same level for less client users.
- But the results shows that VM-based cloud platform such as XenServer is superior for large amounts of client users.
- However, Windows Azure is much easier implemented than XenServer.
- But the XenServer is Open Source Software(OSS) except enterprise version.

Conclusions

- The experiment results indicate that, from a performance perspective, a private cloud platform can fulfill the requirements if there are a small number of users. Virtualization is the better solution to satisfy a large number of users accessing the server.

Acknowledgement

- The authors would like to thank the National Science Council of the Republic of China for financially supporting this research under Contract No. NSC 100-2221-E-166 -008 -, and MOEA project 99-EC-17-A-03-S1-005.

THANK YOU FOR YOUR LISTENING.

Q & A