

# BUILDING AND MANAGING LARGE SCALE DISTRIBUTED SERVICES

By: Jin Liang

University of Illinois at Urbana-Champaign, 2007

Summarized and reviewed: Mohammad Al-Mutawa  
University of Colorado-Boulder

# Introduction

- Many large scale distributed applications that provide various Internet scale services have emerged recently. Examples of the services include content distribution, storage service, and name service.
- These services often involve large number of application nodes that collaborate with each other in a decentralized fashion.

# Introduction (cont.)

- Potentially those services can achieve high scalability, availability, reliability and QoS/performance.
- However, building large scale service applications is a challenging task since we have to deal with wide area environment (Internet) which makes deploying, monitoring and controlling the application very difficult.

# The Goal

- The goal is to design and develop techniques aimed at simplifying the *design, implementation and management* of large distributed applications.

# Background

- There are two classes of large distributed applications:
- *Infrastructure* based application.
- *End host* based application

# Background (cont.)

- Infrastructure based applications are often provided and managed by a single organization. They consist of a large number of application nodes (100's to 10,000's) .
- The application nodes are symmetric in terms of functionality and they communicate with each other in a decentralized fashion.
- Normally used for content distribution, name services, and storage services

# Background (cont.)

- End host based applications are also called peer-to-peer (P2P) applications.
- P2P applications are not provided and managed by a single entity.
- P2P are instable as end users can join and leave very frequently.
- Examples : Bit Torrent, KaZaA

# Approach

- For the design phase:  
Liang introduces a layered architecture called OCMA (Overlay Construction and Maintenance Architecture)
- OCMA divides such applications into three layers: the membership layer that keeps up-to-date information about other nodes in the system; the overlay layer that builds or maintains the desired overlay structure; and the application specific layer that carries out application specific processing.



# Approach

- For the Implementation phase:  
Liang presents a C++ framework called PPF (Protocol Plug-in Framework) that can be used to implement service applications.
- PPF simplifies application implementation because it frees the application developer from the details of asynchronous network programming, and it allows the same code to run in both simulation and real world mode.

# Approach (cont.)

- For the management phase:  
They present 2 tools:
  1. MON (Management Overlay Networks)  
Provides dynamically query and control
  2. InfoEye  
A self-configuring information monitoring system