Overlay and P2P Networks

Introduction

Prof. Sasu Tarkoma

13.1.2014
Contents

• Course Overview
• Lectures
• Assignments/Exercises
Course Overview

• Overlay networks and peer-to-peer technologies have become key components for building large scale distributed systems.

• This course will introduce overlay networks and peer-to-peer systems, discuss their general properties, and applications. The course will cover the following topics:
  – Currently deployed peer-to-peer systems and how they work
  – Distributed Hash Tables as a base for structured peer-to-peer systems
  – Peer-to-peer storage systems
  – Performance issues, legal aspects, and privacy issues
  – Peer-to-peer content distribution algorithms
General Info

Advanced course, 4 credits

The course replaces the P2P Networks course
You cannot take this course if you took the old course

Requirements: basics of networking

Assignments/exercises done as group work (1-2 persons),
idea is to keep the same group structure (but do tell about freeriders!)
Lectures

• Lectures

• Assignments/exercises
  – Assignment topic given two weeks before deadline

• Course based on book
Support material

Lectures

16.1. Unstructured networks I
20.1. BitTorrent and unstructured networks II
23.1. Modelling and evaluation (with Petri Savolainen)
3.2. Distributed Hash Tables (DHTs) I
6.2. DHTs II
10.2. Applications I
13.2. Applications II (also invited speakers)
17.2. Conclusions and summary
Grading

Course grading will be based on the final exam and the assignments/exercises.

Course exam 26.2.2014 16:00 B123

Final exam on 25.4.2014 16:00 in B123
Assignments/Excercises

- Assignments are given two weeks before the session, due date is the day before the assignment session 4pm.
- Assignments are done in two-person groups (or alone), groups can change between assignments.
- Assignments give bonus points for the exam:
  - Max exam points 18, max bonus points 4
- Wednesdays 16-18 B222 last session 19.2.
  - 15.1. Reception on questions I
  - 22.1. Answers to questions I
  - 29.1. Reception on questions II
  - 5.2. Answers to questions II
  - 12.2. Reception on questions III
  - 19.2. Answers to questions III
Schedule summary

13.1. Introduction. Exercises. (Exercise I published)
15.1. Exercises: Reception on questions I
16.1. Unstructured networks I
20.1. BitTorrent and unstructured networks II
22.1. Exercises: Answers to questions I (Exercise II published)
23.1. Modelling and evaluation
29.1. Exercises: Reception on questions II
3.2. Distributed Hash Tables (DHTs) I
5.2. Exercises: Answers to questions II (Exercise III published)
6.2. DHTs II
10.2. Applications I
12.2. Exercises: Reception on questions III
13.2. Applications II (also invited speakers)
17.2. Conclusions and summary
19.2. Exercises: Answers to questions III
Consistent hashing alleviates network problems and eventual consistency can be achieved through replication and synchronization.

Examples: Dynamo, Cassandra

Replication, Gossip, etc.

Selective flooding

Consistent hash (O(1) DHT)

Good for arbitrary data and search functions, can aggregate routing info, structure improves scalability

Examples: Gnutella and Freenet

Example: BitTorrent

Limited flooding / depth first / Bloom filters

Tracker

DHT

Examples:
Lookup: Chord, CAN, Kademlia
Storage: PAST
Rendezvous: Scribe (for multicast), i3

Wide-area (structured)

Wide-area (unstructured)

Cluster

Search
Storage
Rendezvous
Search
Storage
Rendezvous
Search
Storage
Rendezvous

Internet (TCP/IP)

Good for name/value data, note flat address space, one node is responsible, churn is a concern

Examples:

 look-up: Chord, CAN, Kademlia
 storage: PAST
 rendezvous: Scribe (for multicast), i3

Good for arbitrary data and search functions, can aggregate routing info, structure improves scalability

Examples: Gnutella and Freenet

Example: BitTorrent

Limited flooding / depth first / Bloom filters

Tracker

DHT

Examples:
Lookup: Chord, CAN, Kademlia
Storage: PAST
Rendezvous: Scribe (for multicast), i3

Wide-area (structured)

Wide-area (unstructured)

Cluster

Search
Storage
Rendezvous
Search
Storage
Rendezvous
Search
Storage
Rendezvous

Internet (TCP/IP)

Good for name/value data, note flat address space, one node is responsible, churn is a concern

Examples:

 look-up: Chord, CAN, Kademlia
 storage: PAST
 rendezvous: Scribe (for multicast), i3

Good for arbitrary data and search functions, can aggregate routing info, structure improves scalability

Examples: Gnutella and Freenet

Example: BitTorrent

Limited flooding / depth first / Bloom filters

Tracker

DHT

Examples:
Lookup: Chord, CAN, Kademlia
Storage: PAST
Rendezvous: Scribe (for multicast), i3

Wide-area (structured)

Wide-area (unstructured)

Cluster

Search
Storage
Rendezvous
Search
Storage
Rendezvous
Search
Storage
Rendezvous

Internet (TCP/IP)
<table>
<thead>
<tr>
<th>Main theme</th>
<th>Prerequisites</th>
<th>Approaches learning goals</th>
<th>Meets learning goals</th>
<th>Deepens learning goals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overlay and peer-to-peer networks: definitions and systems</td>
<td>Basics of data communications and distributed systems (Introduction to Data Communications, Distributed Systems)</td>
<td>Knowledge of how to define the concepts of overlay and peer-to-peer networks, and state their central features</td>
<td>Ability of being able to compare different overlay and p2p networks in a qualitative manner</td>
<td>Ability to give one’s own definition of the central concepts and discuss the key design and deployment issues</td>
</tr>
<tr>
<td>Distributed hash tables</td>
<td>Basics of data communications and distributed systems (Introduction to Data Communications, Distributed Systems) Big-O-notation and basics of algorithmic complexity</td>
<td>Knowledge of the concepts of structured and unstructured networks and the ability to classify solutions into these two categories Knowledge of the basics of distributed hash tables Ability to describe at least one distributed hash table algorithm in detail</td>
<td>Ability of being able to compare different distributed hash table algorithms Ability of designing distributed hash table-based applications Knowledge of key performance issues of distributed hash table systems and the ability to analyze these systems</td>
<td>The knowledge of choosing a suitable distributed hash table design for a problem Familiarity with the state of the art</td>
</tr>
<tr>
<td>Reliability and performance modelling</td>
<td>Basics of probability theory Basics of reliability in distributed systems</td>
<td>Ability to model and assess the reliability of overlay and peer-to-peer networks by using probability theory Knowledge of the most important factors pertaining to reliability</td>
<td>Ability of analytically analyzing the reliability and performance of overlay and peer-to-peer networks Understanding of the design issues that are pertinent for reliable systems</td>
<td>Familiarity with the state of the art</td>
</tr>
<tr>
<td>Content distribution</td>
<td>Introduction to Data Communications</td>
<td>Knowledge of the basic content distribution solutions Ability to describe at least one overlay and p2p network based content distribution solution</td>
<td>Knowledge of different content distribution systems and the ability to compare them in detail Knowledge of several content distribution techniques</td>
<td>Familiarity with the state of the art</td>
</tr>
<tr>
<td>Security</td>
<td>Basics of computer security</td>
<td>Knowledge of the basic security issues with overlay and p2p networks Knowledge of the sybil attack concept</td>
<td>Ability to discuss how security problems and limitations can be solved Knowledge of how to prevent sybil attacks</td>
<td>Knowledge of how to prevent sybil attacks Familiarity with the state of the art</td>
</tr>
</tbody>
</table>
Contact information

Lecturer prof. Sasu Tarkoma (contact info on homepage)

Assignments: Juhani Toivonen (@cs.helsinki.fi)

Course homepage can be found: www.cs.helsinki.fi/courses
Questions?