Course overview

Johan Montelius

KTH

HT15

Registration

Register by signing your name on the list, this or next week. If your name is not on the list:

- you have not done course selection,
- you have not done registration for the semester ("terminsregistrering") o
- they have not registered you for the the program ("programregistrering") or
- something else.

If your name is not on the list you will not be registered on the course (and even if it is does not guarantee that you will be).

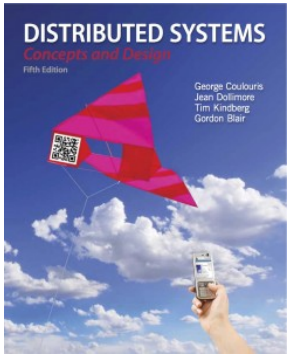
Course goal

You should after the course be able to:

- explain important characteristics of distributed systems
- describe architectural and fundamental models of distributed systems
- explain and compare strategies for inter-process communication
- explain and compare middleware models
- explain and compare name services
- explain the concept of logical time
- use logical time to implement distributed algorithms

Literature

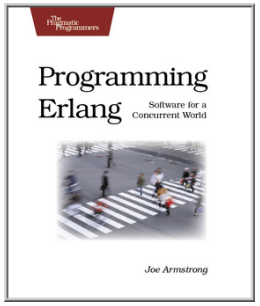
- "Distributed Computing - Concepts and Design",
- 5'th edition (4'th ok)
- Coulouris et al,
- Addison Wesley (www.cdk5.net)



Erlang

- "Erlang Programming",
- Francesco Cesarini and Simon Thompson
- O'Reilly

- "Programming Erlang"
- Joe Armstrong
- Pragmatic Programmer



Lectures

Fourteen lectures that will mostly follow the course book. Do read in advance!

Erlang is only given one lecture, you're expected to pick up a new language on your own.

Slides will be available on the web.

Lectures

- 1: Introduction - what is a distributed systems and why is it different. chapter 1 and 2.
- 2: Erlang - concurrent and distributed programming in Erlang.
- 3: Networks and process communication - things you should know but we'll go through them again. chapter 3 and 4

Lectures

- 4: Remote invocation - language constructs to program distributed systems. Chapter 5
- 5: Indirect Communication - group communication, publish/subscribe and message queue systems. Chapter 6
- 6: File systems and Name services - the problems of a distributed file system, performance, consistency chapter 12 and 13

Lectures

- 7: Time - a simple thing that turns out to be very complex. Chapter 14.1-4
- 8: Global state - can we describe the state of a distributed system and what can we determine. Chapter 14.5
- 9: Coordination and agreement - how do we agree and how do we know that we do agree? chapter 15

Lectures

- 10: Transactions - how can we make a set of operations behave as an atomic operation? chapter 16
- 11: Distributed transactions - now how do we solve it if we have multiple servers. chapter 17
- 12: Replication - building fault tolerant systems, chapter 18

Lectures

- 13: Distributed Hash Tables - why do hashing? chapter 10
- 14: Summary and the price of olive oil

Seminars

First session - help with completing the tasks. Not compulsory.

Following sessions:

- hand in written report on how you solved the problem
- be prepared to present your solution
- connect the systems and do some experiments

Select which group to join in Daisy.

Seminars

- Erlang - not compulsory
- Rudy - a small web server
- Routy - message routing
- Loggy - logic time logger
- Groupy - group communication
- Chordy - a distributed hash table

Examination

- compulsory lab session / seminars
  - complete tasks in advance
  - signing the list is "yes I've done it"
  - don't turn up unprepared
  - if you can not attend, email before the seminar
- written examination, closed book
  - A : declarative (multiple choice questions, 24p)
  - B : compare, describe (8 questions, short answers, 16p)
  - C : analytic, reflect (3 questions, essay answers, 12p)

Grading

The first part will, scoring 16 or higher, give you an E.

Given a good result on the first part (aprx 20 points), the second part could give you a D or C.

Given a good result on the first and second part (aprx 22 and 12 points) the third part can give you a B or an A.

Final grade is based on written exam, written reports and active participation in seminar sessions.