Vrije Universiteit Brussel

Proposal bachelor thesis

Title: A library for replicated state machines in AmbientTalk

Promotor: Prof. Dr. Elisa Gonzalez Boix (advisors: Kevin Pinte, Florian Myter)

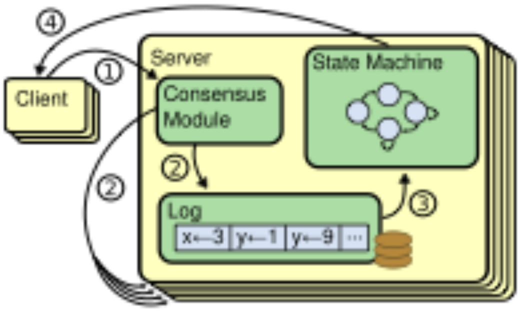
Includes preparation course: **Yes** / No (Select the correct option)

Context

In distributed systems a fundamental problem is to make a collection of nodes (machines) work as a coherent group, even in the case of failures. Often this involves the nodes reaching agreement on a certain value needed for computation. Consensus algorithms solve this exact problem and are therefore a crucial asset in developing distributed systems.

One instance of the consensus problem is the replicated state machine problem. In this context state machines that run on a collection of machines (servers) compute identical copies of the same state. The state machines can even continue to run even if some of the machines fail.

Replicated state machines are typically implemented using a replicated log. Using a consensus algorithm on the servers it is possible to agree over a replicated log. Each server stores a copy of the log which contains a series of commands that originate from clients. The state machine on the server executes the commands from the log in order. Each log contains the same commands in the same order. Since the state machines are deterministic and execute the same commands, they compute the same state and the same sequence of outputs.



Proposal bachelor thesis

At the Software Languages Lab we have developed a high-level distributed programming language, called AmbientTalk, specifically conceived to develop peer-to-peer applications that operate in mobile ad hoc networks. Such networks suffer from many disconnections and peers frequently come and go.

Many of the applications developed in AmbientTalk are structured as state machines and since we deal with distributed applications, consensus is often needed. However, AmbientTalk has no support for implementing state machines or integrate consensus in applications.

The goal of this thesis is to integrate in a high-level language like AmbientTalk, support for consensus algorithms. The student will use AmbientTalk to (1) develop a small library for implementing state machines, and then (2) implement the RAFT consensus algorithm on top of it.

Preparatory course bachelor thesis

(only if applicable)

The student needs to get familiarized with AmbientTalk and basic concepts of distributed systems and consensus algorithms. To this end, he will need to do some literature study based on the following references:

- AmbientTalk: https://soft.vub.ac.be/amop/
- Introduction to Distributed Programming
  http://dl.acm.org/citation.cfm?id=1137759 (book available at SOFT)
- Consensus algorithms and replicated state machines:
  - Raft:
    https://ramcloud.stanford.edu/wiki/download/attachments/11370504/raft.pdf
  - Paxos: http://research.microsoft.com/en-us/um/people/lamport/pubs/paxos-simple.pdf

Contact

Don't hesitate to contact us for further info:
Kevin Pinte kevin.pinte@vub.ac.be 10F723
Florian Myter fmyter@soft.vub.ac.be office 10F732
Elisa Gonzalez Boix egonzale@vub.ac.be office 10F740