IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 2017

MEng Honours Degrees in Computing Part IV
MSc in Advanced Computing
MSc in Computing Science (Specialist)
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the
Associateship of the City and Guilds of London Institute

PAPER C410H

SCALABLE DISTRIBUTED SYSTEMS DESIGN

Tuesday 21 March 2017, 11:40
Duration: 70 minutes

Answer TWO questions

Paper contains 3 questions
Calculators not required

1 a Briefly explain each of the following concepts, and give an example where each could be applied in the context of scalable distributed systems.

i) Resilient distributed dataset (RDD)

ii) Scalability

iii) Gossip-based communication

iv) Stateless service

b Spanner is a globally-distributed database system.

i) Describe the guarantee of *external consistency* that Spanner provides.

ii) State the *TrueTime API* used by Spanner, briefly describing the semantics of each API call.

iii) Describe a proposal how the TrueTime API could be implemented.

iv) Explain what happens immediately after a Spanner deployment has experienced a *network partition* between data centres.

v) Assume that the network partition between data centres remains for a prolonged period of time. How is the Spanner deployment affected? Justify your answer.

vi) Assuming that the TrueTime uncertainty interval, $latest - earliest$, is always between $k\,\mathrm{ms}$ and $2k\,\mathrm{ms}$, in the **best** case, what is the shortest transaction commit time for a write transaction in Spanner? State your answer as an inequality involving k and briefly explain it.

The two parts carry, respectively, 20% and 80% of the marks.

2 a Briefly explain each of the following concepts, and give an example where each could be applied in the context of scalable distributed systems.

i) Location transparency

ii) Batch processing system

iii) Partition/aggregation pattern

iv) Multi-version store

b The Dynamo key/value store follows a decentralised design.

i) Briefly describe, with the aid of a diagram, how Dynamo uses a ring organisation of nodes to support read and write requests in a decentralised fashion.

ii) Identify **two** aspects of Dynamo's design, together with a brief explanation, that are made more challenging due to its decentralised design.

iii) Explain how the Dynamo design could be changed to realise the two aspects identified under (ii) using *centralised* components. Make sure that your updated design does not compromise the scalability of Dynamo.

The two parts carry, respectively, 20% and 80% of the marks.

3 You work as a software engineer for DeepThought, a start-up company that develops new machine learning applications.

Your boss asks you to design a new machine learning platform. The platform should process a large amount of training data (on the order of terabytes) and use it to train different complex machine learning models, which are specified by data scientists employed by the company.

The platform should execute on a cluster of machines. The output should be a trained machine learning model, whose training may take weeks given the available hardware resources. You may assume that the training data is stored on a distributed file system, such as GFS or HDFS.

The design of the new platform should satisfy the following requirements:

(R1) The platform should be easy to use by data scientists who do not have sophisticated programming skills. Data scientists should still be able to realise different machine learning models.

(R2) The platform should support heterogeneous hardware for the training of machine learning models, in particular GPUs and other custom hardware accelerators.

(R3) The platform should be incrementally scalable.

(R4) The platform should be fault-tolerant.

(You should make justified decisions about any other requirements that are left unspecified.)

a Explain why using the existing data processing platforms such as MapReduce or Spark do not satisfy all requirements above.

b Describe the API (or programming model) that data scientists could use to interact with the platform, satisfying requirement (R1).

c Draw a diagram of your *system design*, clearly labelling all distributed components. For each component, explain its operation and justify its function.

d For each of the requirements, (R2)–(R4), explain how your design achieves it.

The four parts carry, respectively, 15% 20%, 35%, and 30% of the marks.