

BOOKWORK 2

Zach Moolman

QUESTION 1

There are a couple of things that we can do.

1. We can look at the protocol that is used for the application. We might want to connectionless protocol that use less bandwidth.
2. We can consider how we can scale the operation. There are two schools of thought.
 1. we can add bandwidth by vertically scale the hardware but light can only travel so fast.
 2. As of late the focus is to scale horizontal by increasing the capacity by adding more nodes(Machines) to the network. This will allow us to share the load between the nodes. There are different approaches and is the main focus of this course.

QUESTION 2

1. Vertical Distributed Systems We place logical different component on different machines. The main idea is that you can split logical units out such as databases, user interface, or processing component onto different machines. This, has

become a feature of NoSQL types of distributed databases where the data can be split by columns and stored on different machines.

2. Horizontal Distributed Systems It is the number of the servers and clients that counts. The servers and clients are split up into segments but each work on its own share of work. This is often referred to as load balancing. Example of this is peer-to-peer systems.

Yes, for example databases are a logical unit that can run as a logical unit on a separated machine but the same time we can have a distributed database working only on there part of a distributed database.

QUESTION 3

I can think of breaking systems in cars. All wheels can work(break) independently but as a coherent systems they can work together to only apply certain amount of break as required. This will required that they share information among them self.

1. Each break can act autonomously
2. The user sees the car break system as one system
3. Information is shared
4. fault-tolerate is required as part of a critical system such as breaks

QUESTION 4

Three-tiered architecture expands on the client server architecture. Sometimes, we forget that the server might act as a client. An typical use for three-tiered architecture is used in transaction processing and website.

The client connects to the web server , which in turn connects to a database or an application server and then respond back the client.

Client

```
-- |
  | Request from client operation to  webservber
  ----->|
        | Request from webrserver to database
        |----->|
              |Database server
        |<-----|
        |Return data from database to webserver
  <-----|
  |Return processed web request from webserver
-- |
```

QUESTION 5

Inceptor is software construct that will break the normal flow of control and allow other code to be executed. Inceptor is primary means for adapting middleware to the specifics needs of an application. To implement a inceptor is not always an easy task and take some effort. Therefore, it can be difficult to maintain Inceptor if the target environments keeps on changing. A solution to to this is the construction of adaptive software. The idea is that the middleware is adaptable including to allow to change the inceptor without bringing the whole system down. In other words we want to change software component at rum time.