TECHNISCHE UNIVERSITEIT EINDHOVEN
Faculteit Wiskunde en Informatica

Examination Architecture of Distributed Systems (2IMN10),
on Thursday, November 7, 2019, from 9.00 to 12.00 hours.

Before you start, read the entire exam carefully. Answers to all questions must be motivated and stated clearly. For each question the maximum obtainable score is indicated between parentheses. The total score sums up to 20 points. This is a closed book exam, i.e., you are not allowed to use books or other lecture material when answering the questions.

1. (2 points) Describe the peer-to-peer architectural style using the appropriate vocabulary. Name the concepts and rules involved, give a motivation for its usage, and mention typical behavior and its weak points.

   Answer. See slide 15 of the slide set on architectural styles.

2. An essential ingredient of many architectural styles is a discovery service.

   (a) (0.5 point) Describe the problem addressed by service discovery and its solution in generic terms and indicate common architectural elements that are involved.

   Answer. Service discovery addresses the problem of how two parties (a *service seeker* and a *service provider*) that do not know each other can find each other. To that end, the service providers publish *advertisements* and the service seekers issue *queries*, that specify the service either by name or through desired attribute values. These are propagated through the system and compared at certain locations possibly with the assistance of a third party called a *mediator*. The lists of matching service instances is presented to the seeker which either selects one, or rejects them all and issues a new query.

   (b) (1 point) There are two fundamentally distinct ways in which to perform service discovery. Name both approaches and describe them in some detail. Mention advantages and disadvantages of both.

   Answer. The two approaches are

   Mediated discovery In this style the seeker queries a known lookup service maintained by a mediator (broker) which contains a repository of service advertisements. For each service, its access point and possibly a set of other attributes is maintained. These are communicated to the seeker, which subsequently accesses the service. It is also possible that the repository only holds the IP-address of the machine that hosts the service and that this host runs a deamon that needs to be queried for the true access point of the service.

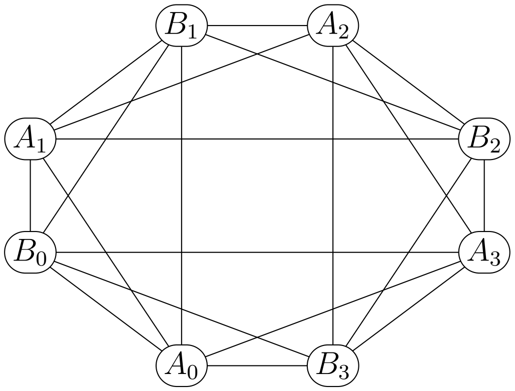
Immediate discovery In this style queries and/or advertisements are broadcasted (multicasted) within a network, and matching is done at either the seeker's or the provider's side.

The advantages of one style are the disadvantages of the other. One disadvantage of immediate discovery is that both parties need to be aware of the complete protocol and maintain corresponding state (e.g. a potential seeker may listen continuously on a network and thereby assemble a list of service instances). Another disadvantage of immediate discovery is that it is limited in scope (the broadcast/multicast domain), whereas mediated discovery can be global. Finally, immediate discovery is less scalable than mediated discovery, because of the growth of bandwidth usage and response time due to multicasting. An advantage of immediate discovery is that it is fully distributed, requires no additional configuration (zeroconf) of the system by a third party, such as a system administrator installing, e.g., name servers.

(c) (1 point) For each approach, give an example of a concrete service that applies it, i.e., give both the service's name and the type of service provided.

Answer. DNS uses mediated discovery to resolve domain names to IP-addresses. Corba uses mediated discovery, by means of ORBs (object request brokers) to resolve object references, i.e., to access remote objects. Immediate discovery is used by protocols such as DHCP that is used by host machines to acquire an IP-address on a local net. Another example is ARP that is used by routers to resolve an IP-address to a local link address. Also UPnP uses immediate discovery.

3. Consider a replicated distributed data store with 8 replicas each managed by an individual replica manager (RM). For communication within the data store, the RMs are connected according to the following network topology.



Clients access the data store via an RM. Each client always contacts a single RM, but may contact distinct RMs in successive operations. To increase availability under network partitions, while maintaining eventual consistency, the data store uses Gifford's quorum protocol, with read quorum size NR and write quorum size NW.

(a) (1.0 point) Explain how Gifford's quorum protocol works and indicate the constraints that need to be imposed on the quorum sizes in order for the protocol to achieve consistency.

Answer. To execute an operation the data store needs to establish a subset of RMs, called a quorum, that is capable to engage in the operation, i.e., is reachable from the RM where the operation is submitted to the store. For read operations, the size of the set is given by NR, and for write operations by NW. Upon writing, the data object is updated in all replicas of the write quorum and is given a unique time stamp (version number) that is more recent than any time stamp handed out in an earlier update. Upon reading the value returned to the clients is the value from the replica that holds the most recent time stamp. For a data store with N replicas, the quorum sizes need to satisfy two constraints:

- $NR + NW > N$, to prevent read-write conflicts,
- $NW > N/2$, to prevent write-write conflicts.

(b) (0.5 point) A protocol for accessing a given replicated distributed data store with at least one correct node is *t-read-resilient* (*t-write-resilient*), when, in the presence of at most t faulty nodes, all clients that contact a correct node can perform a read (write) operation. Determine both the maximum read-resilience and the maximum write-resilience (maximum values of t), in case this data store uses a quorum-based protocol with $NR = 1$ and $NW = 8$. Assume that a faulty node is also incapable of performing routing actions necessary to assist a correct node in assembling a quorum.

(c) (0.5 point) The same question for $NR = 2$ and $NW = 7$.

(d) (0.5 point) The same question for $NR = 3$ and $NW = 6$.

(e) (0.5 point) The same question for $NR = 4$ and $NW = 5$.

Answer. To answer these questions, we observe the following structural properties of the network.

i. If there are t faulty nodes, then there are $N-t$ correct nodes. So, with $N = 8$, a correct node can assemble a quorum (both for reading and writing) of at most $8-t$ nodes.

ii. Each node has exactly 5 neighbors. Hence, with $t > 4$ faulty nodes, a correct node can become isolated and therefore can only assemble a quorum consisting of 1 node, namely itself. Moreover, with $t \leq 4$ faulty nodes, each correct node can reach a neighbor, and therefore assemble a quorum of at least 2 nodes.

iii. Each node lies on one of two cycles, the one consisting entirely of A-nodes, the other entirely of B-nodes. Moreover, each node has 2 neighbors on its own cycle and 3 neighbors on the other cycle. Hence, if $t \leq 3$ nodes are faulty, there is one cycle with at least three correct nodes left, which are connected and form a line graph. The remaining correct nodes in the other cycle are connected to at least one node of this line. Hence, the network spanned by the correct nodes is connected and therefore each correct node can assemble a quorum of size $8-t \geq 5$.

iv. If $t = 4$ nodes are faulty, then it may happen that the network partitions in two connected subnetworks, each of size 2. E.g., when nodes $A0, B0, A2$, and $B2$ are faulty, the network is split in parts $\{A1, B1\}$ and $\{A3, B3\}$, with an edge between the nodes in each part. Hence, each correct node can assemble a quorum of 2 nodes.

Combining these observations we find that for

$NR = 1$, $NW = 8$ the protocol is 7-read-resilient and 0-write-resilient,
$NR = 2$, $NW = 7$ the protocol is 4-read-resilient and 1-write-resilient,
$NR = 3$, $NW = 6$ the protocol is 3-read-resilient and 2-write-resilient,
$NR = 4$, $NW = 5$ the protocol is 3-read-resilient and 3-write-resilient.

4. Informally, any consensus protocol decides on a value v based on proposals made by each of its participants.

(a) (1 point) More formally, a consensus protocol needs to realize the following four properties: termination, validity, integrity, and agreement. Give the definition of each of these properties.

Answer.

Termination: Every correct process eventually decides some value.

Validity: If a process decides v, then v was proposed by some process.

Integrity: No process decides twice.

Agreement: No two correct processes decide differently.

(b) (1 point) Give at least two examples of applications that require a consensus protocol. For each case indicate what the entity is that they need to decide on.

Answer. Any system that uses a blockchain to maintain a ledger requires a consensus protocol. The protocol decides on the next block to be added to the ledger. A totally ordered multicast needs consensus on the next message to be delivered to all correct processes. A distributed transaction system needs consensus on whether to abort or commit on a transaction.

(c) (0.5 point) Besides regular consensus there also exists a variant that is called uniform consensus. Which of the four properties mentioned above is different and what is this difference?

Answer. Consensus and uniform consensus differ in their agreement property. Uniform consensus requires uniform agreement, meaning that all processes that decide should agree on their decision, not just the correct processes. So, for instance a process may decide and subsequently crash, forcing the remaining correct processes to go along with its decision.

5. Consider the Chord scheme for DHTs. Assume a 7-bit identifier space, and assume that the node set N is given by $id(N) = \{8i \mid 0 \leq i < 16\}$.

   (a) (0.5 point) Give the finger table of node 32.

   Answer. For a 7-bit identifier space all finger tables have 7 entries. Table FT_{32} is given by:

   $FT_{32}[1] = FT_{32}[2] = FT_{32}[3] = FT_{32}[4] = 40$, $FT_{32}[5] = 48$, $FT_{32}[6] = 64$, and $FT_{32}[7] = 96$.

   (b) (1.0 point) What is the maximum number of steps necessary to resolve a key? For your answer, you may assume that each node is aware of the identity of its predecessor and, consequently, resolves all keys for which it is responsible in zero steps.

   Answer. Each time the resolution process forwards the lookup query from node n to node $(n + x) \bmod 2^7$, the increment x satisfies $x \in \{8, 16, 32, 64\}$. Furthermore, in any sequence of forwarding operations, the increments are decreasing. For $x = 64$ the next increment is smaller, because otherwise the lookup query travels at least once around the ring (since there are only $2^7 = 2 * 64$ identifiers) which cannot be the case. For $x \in \{8, 16, 32\}$, the next increment is smaller, because otherwise the query would have been forwarded from n to $(n+2x) \bmod 2^7$. Since there are at most 4 distinct increments, the maximum number of steps is therefore 4.

   (c) (0.5 point) Give a key that requires the maximum number of steps to be resolved, when starting the resolution at node 32. Indicate the nodes that are visited when resolving the given key.

   Answer. Starting at node 32, any key k with $17 \leq k \leq 23$ will need 4 steps to be resolved. For $k = 17$ the path traversed is $32 \rightarrow 96 \rightarrow 0 \rightarrow 16 \rightarrow 24$. Note that the algorithm described in TvS also takes 4 steps for key 24, but since node 32 knows that node 24 is its predecessor, it could take advantage of that knowledge and forward the query directly to its predecessor.

   (d) (0.5 point) Describe the adaptations that have to be made to the DHT when node 32 leaves the set of nodes N. You may assume that the node leaves voluntarily and, hence, will cooperate in the realization of the required changes.

   Answer. Node 32 notifies all other nodes that it will leave the set and what its successor node(40) and predecessor node(24) were. All nodes adapt their finger tables, replacing the entry 32 (if it occurs) by 40. In addition, node 40 makes node 24 its predecessor. Note that, since $FT_n[1]$ always points to the successor of n, node 24 will in fact adjust its successor by modifying its finger table. Furthermore, notice that node 32 need not be aware of the identity of all nodes in N, so notification may require the assistance of additional nodes, e.g., by forwarding the notification along the ring of nodes in a clock-wise direction. Of course, all data associated with keys $25, \ldots, 32$ has to moved to node 40 as well, but that does not involvethe DHT

6. (1.5 points) Describe in detail how the URL

   ```
   http://www.win.tue.nl/home/wsinmak/Education/2IMN10/ADS.html
   ```

   is resolved. In particular, indicate the closure mechanisms, and resolution procedure for the various parts of the URL.

   Answer. This URL consists of three components: a scheme http that names a protocol, a domain name www.win.tue.nl that identifies the host holding the resource, and a pathname home/wsinmak/Education/2IMN10/ADS.html that identifies a file (the resource) on that host. It is resolved as follows. The browser extracts the scheme and based on the value found invokes its http-client, which is a plugin in the form of a library. This http-client resolves the hostname by contacting its local DNS-server to obtain an IP-address. The local DNS-server may invoke other DNS-servers to assists in resolving the hostname (see TvS for a description of how this works), but in case the local DNS-server happens to be the TU/e DNS-server it will know the IP-address. Next, the Http client will set up a connection with the host machine, construct an appropriate http-request (GET) containing the pathname and invoke the operating system to send the request to the web server at the host machine. Finally, this web server resolves the pathname to obtain the resource.

7. Indicate for the following statements whether they are true or false. Motivate your answer with a short argument.

   (a) (0.75 point) Erasure codes make it possible to reconstruct a k symbol message encoded by n symbols, if any $k' < n$ of the encoded symbols have been received. Erasure correction is a form of backward error recovery.
   Answer. False.
   Backward error recovery returns the system to a previous correct state, i.e., before the message whose symbols got lost was sent. Erasure correction, however, brings the system *forward* to a correct state in which the message is reconstructed as if it were correctly received.

   (b) (0.75 point) A brokered publish-subscribe architectural style supports modifiability.
   Answer. True.
   Publish-subscribe provides referential decoupling. Hence, e.g., protocol changes on either the publisher or subscriber side only affect the brokers.

   (c) (0.75 point) Susan needs to design a system that maintains topic-based newsgroups. To ensure that subsequent news items on a topic make sense to their readers, she should implement her system such that it guarantees monotonic-read consistency.
   Answer. False.
   Monotonic-read consistency only guarantees that a next read operation by the

same reader sees a later news item. It does not guarantee that all items in between are also seen, which may be needed to make sense of the last item. What is required is writes-follow-reads consistency.

(d) (0.75 point) mDNS + DNS-SD can be used to discover all weather (meteorological) services offered by the various airports in a country.
Answer. False.
mDNS in combination with DNS-SD is used on a local link of a single net, whereas the scenario in the statement involves internet communication.

(e) (0.75 point) Load balancing improves the scalability of a system.
Answer. True.
When the load on a server reaches that server's capacity, the quality of service, most noticeable its response time, starts decreasing. By replacing a single server by a cluster of servers, one would hope to increase the system's capacity to the sum of the capacities of the individual servers. For this to happen, however, queries need to be handled by servers that still have spare capacity. This is exactly what load balancing tries to achieve. Thus, load balancing minimizes the increase in resources needed to accommodate an increase in load, or makes it possible to accommodate a wider range of loads with the same capacity, which are both scalability objectives.

(f) (0.75 point) Proxies provide access transparency.
Answer. True.
Although reasons for their existence vary wildly, all proxies share the fact that they are representations for objects at other locations. They are designed in such a way that clients requiring a service from the original object can obtain that same service in the same way,i.e. using the same interface, from the proxy. So, in terms of interaction, proxies are indistinguishable from the original, which is referred to as access transparency.

(g) (0.75 point) An architectural viewpoint is a collection of models
Answer. False.
A viewpoint is a collection of patterns, templates, and conventions for constructing models that address a specific (set of) concern(s). In addition, a viewpoint identifies stake-holders that have an interest in those concerns. A collection of models constructed according to the principles and guidelines of a particular viewpoint, on the other hand, is called a view.

(h) (0.75 point) A broker is one of the building blocks of message-queuing middleware.

Answer. False.
A broker is a separate process on top of message queues that provides application protocol translation and multi-cast routing. In particular, it forwards messages from a publisher to all of its subscribers, thus realizing referential decoupling.