Norwegian University of Science and Technology
Department of Telematics

Contact during exam [Faglig kontakt under eksamen]:
Bjarne E. Helvik (92667)

EXAM IN COURSE [EKSAMEN I EMNE]
TTM4120 Dependable Systems [Pålitelige systemer]

Thursday [Torsdag] 2007-05-31
09:00 – 13:00

The English version starts on page 2.

Den norske bokmålsutgaven starter på side 10.

Hjelpemidler:
D - No printed or handwritten material is allowed. Predefined simple calculator [Ingen trykte eller håndskrevne hjelpemidler tillatt. Forhåndsbestemt enkel kalulator]

Sensur week 25, 2007

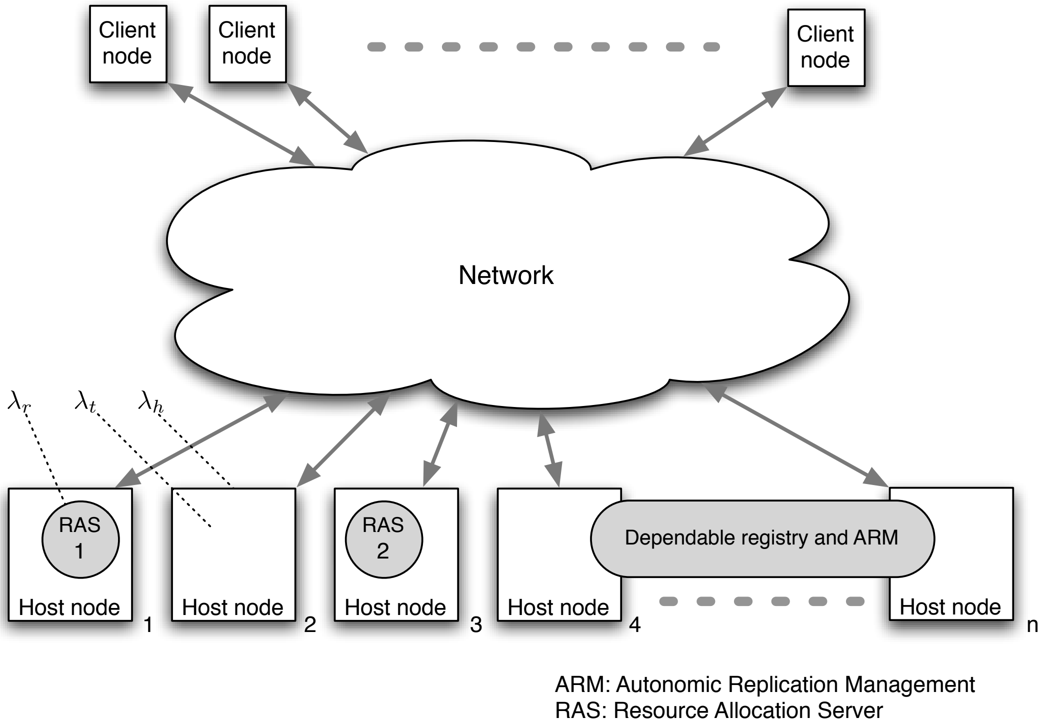


Figure 1: Illustration of system studied.

English version[1]

This exam will use as its basis a system providing a resource allocation service, similar to the one developed and studied by the students during the course. The main difference is that it is extended with an autonomic replication management (ARM) module as presented in the course curriculum. The core functionality of ARM, the replication manager (RM) enables us to extend the failure detection, and to reconfigure the system in case of host node and replica failures, as described in the syllabus. In normal operation, the resource allocation service is provided by two load shared resource allocation servers (RAS). In case of a failure, the remaining server will take the entire load, and, if feasible, the RM will create a new RAS, which will synchronize with the remaining replica and take a part of the load. The RAS and interaction between clients and servers are handled by the Jgroup middleware. For the sake of simplicity, it is assumed that the clients, the network as well as the dependable registry (DR) and the RM do not fail. A sketch of the system is shown in Figure 1. The rate of permanent and temporary host node failures are λ_h and λ_t respectively. A RAS replica fails with a rate λ_r. All failures are independent and are crash failures.

a) Define what is meant by a crash failure. Why do we often design system units to have this

[1] In case of divergence between the English and the Norwegian version; the English version is in force.

behaviour when they fail? What is the term for the most likely (dominating) failure behaviour of a system unit?

A crash failure means the unit does not produce any results when it fails, and do not produce any output/results before explicit action is taken to recover it. It is easier to design a fault-tolerant system if one knows that all results produced are correct and timely, i.e. the "only" kind of failure we have to handle is no results. When this behaviour is consistent it may in some case make the task easier (e.g. to detect that a failure has occurred. (To ease the design of the overall system is a sufficient answer.) Failure semantics.

b) Define verbally the reliability of a system, and give the mathematical definition of the reliability function for a system which is as new when service/system is started. If a system is modelled by a reliability block diagram and this diagram shall be used to predict the reliability function of the system, what are the requirements (assumptions that must be made) for each of the system elements if they are modelled as blocks in the diagram?

- The reliability of a system is its ability to provide uninterrupted service.
- The reliability function $R(t) = \Pr(T_{FF} > t)$ where T_{FF} is the time from the system/service starts as new and until the first failure.
- The requirements are:
  - The system elements modelled as a block fail independently of each other.
  - A failed block/system element is not repaired/restored to operation

c) We are interested in finding the service life time if the system presented in the introduction is put into operation and left without manual maintenance, i.e., when a host node fails it is not repaired. The RM will seek to maintain a working configuration of the service. Short down time periods during reconfiguration are, when dealing with this question, acceptable. The system has n host nodes, each having a failure rate λ_h. Nodes fail independently of each other. Argue very briefly why the requirements mentioned in b) are met. Find expressions for the expected life time of the service, t_L, and the time that the system will survive with a probability of 0.9, $t_{0.9}$.

Replica failures and temporary failures may be neglected. The service, with the assumptions in the text, will work as long as there is at least one host working. Host nodes fails independently of each other and are not paired. Hence, we have a 1-out-of-n (i.e., a parallel) system. The expected life corresponds to the mean time to first failure taking only permanent failures into account.

$$t_L = \mathrm{MTFF}_h = \int_0^\infty (1 - (1 - \exp(-\lambda_h t))^n) dt = \lambda_h^{-1} \sum_{i=1}^{n} \frac{1}{i}$$

Based on the same line of reasoning, we have for the 'permanent failure only' reliability function

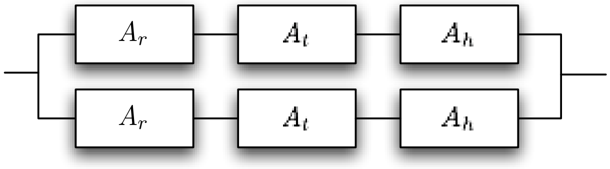
$$R_h(t_{0.9}) = 1 - (1 - \exp(-\lambda_h t_{0.9}))^n = 0.9$$

which solved yields $t_{0.9} = -\lambda_h^{-1} \ln(1 - \sqrt[n]{0.1})$.

In the following, the system will also undergo manual maintenance (repair). The expected manual repair time of a host is μ_h^{-1}. The expected restart time of a host after it has crashed due to a non-permanent fault is μ_t^{-1}. The expected time it takes for the RM and Jgroup to create a new replica on a running host (including the time possibly needed to synchronize with the other replica) is μ_n^{-1}. We have that $\mu_n \gg \mu_t \gg \mu_h$.

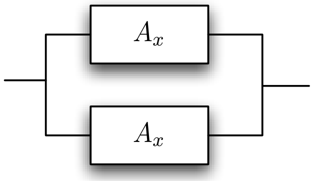
d) Use a reliability block diagram to obtain an approximation of the asymptotic availability of the resource allocation service when the system has the minimal number of host nodes ($n = 2$): A_2. If additional assumptions are necessary, state these..

In this case, a replica cannot become operational before the actual failure is rectified. To obtain an approximation, we assume that all failures are handled independently, and that a host may be divided in an independently failing permanent and a non-permanent component. This yields component availabilities: $A_r = \frac{\mu_n}{\mu_n+\lambda_r}$, $A_h = \frac{\mu_h}{\mu_h+\lambda_h}$,and $A_t = \frac{\mu_t}{\mu_t+\lambda_t}$, These form a series structure for each working replica; the two replicas work in parallel. Hence, $A_2 = 1 - (1 - A_r A_t A_h)^2$.



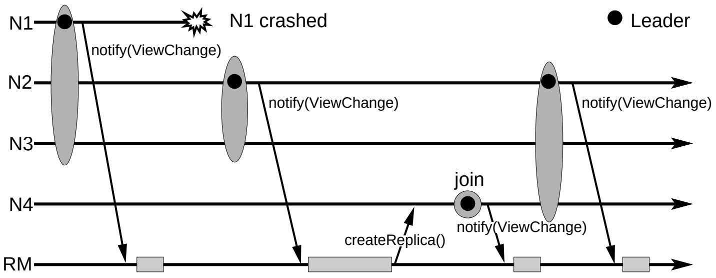
e) Obtain an approximation of the asymptotic availability of the resource allocation service when the system has a large number of host nodes (n approaches infinity): A_∞. (Hint: A reliability block diagram may be used.) If additional assumptions are necessary, state these..

In this case, there will always be a host available on which the RM may create a new replica. Since the replica creation time is shortest, the shortest down-times are obtained by this approach. Hence, the availability of one replica becomes $A_x = \frac{\mu_n}{\mu_n+\lambda_r+\lambda_t+\lambda_h}$. Since the two replicas constitute a parallel system, we have $A_\infty = 1 - (1 - A_x)^2$.

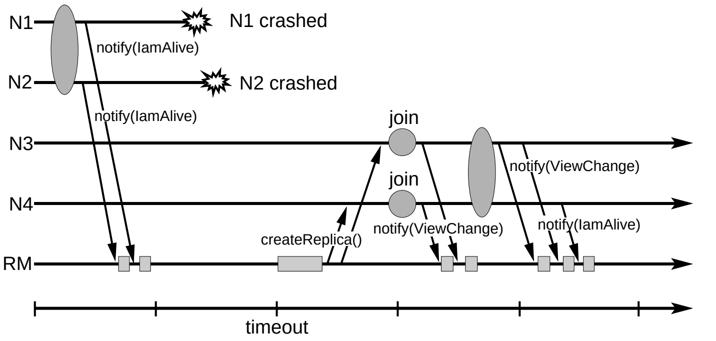


f) In order to create new replicas after failures, ARM must become aware of the failure. Describe two means used in the Jgroup/ARM system for failure discovery.

1) The failure of a member of a group (here the two RAS) will trigger a "ViewChange" in Jgroup. The (new) leader of the group will notify the replica manger (RM) about this. The RM will take the necessary reconfiguration actions. See the figure below.



2) If all members of a group fail before the RM is notified (or the group has only one member), the RM becomes aware of the group failure since "IamAlive" notifications from the replica will be missing. If an "IamAlive" notification does not arrive within a certain time-out period, the RM takes action and create new replicas. See the figure below. (The time-out period may be set dependent on the number of replicas of the group, matching the likelihood of a group failure. Short for single replica groups.)



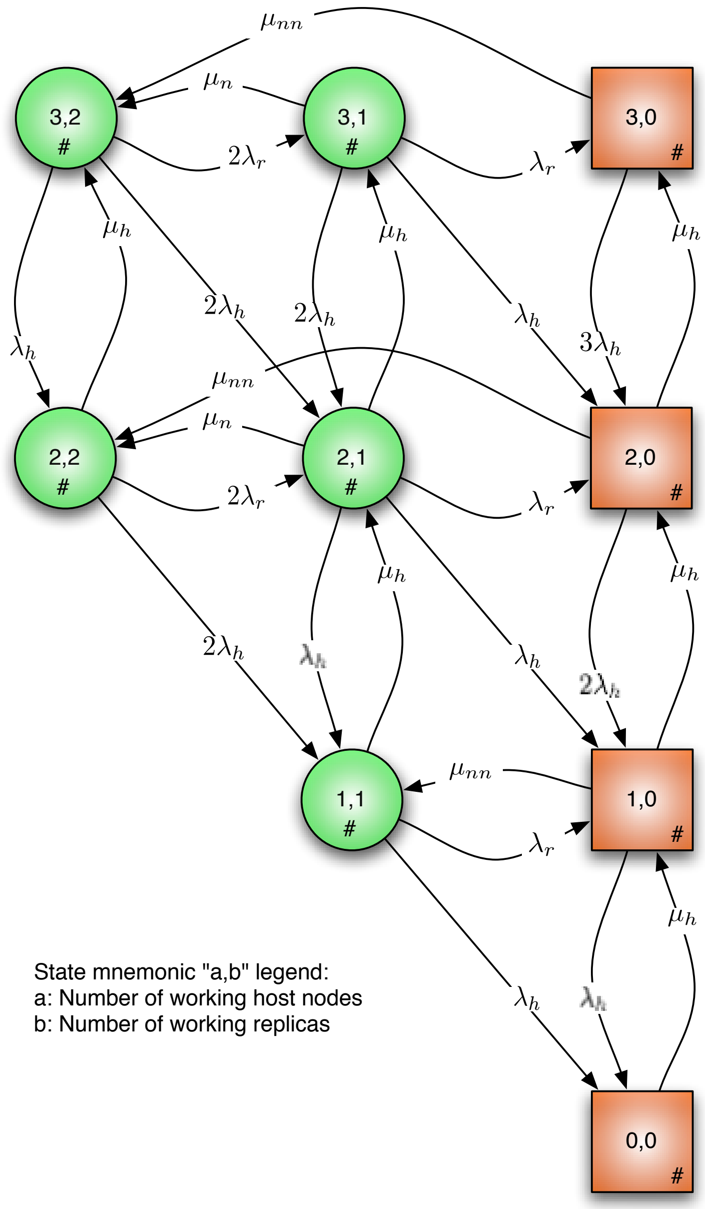
The models of the service availability developed under d) and e) are not sufficiently accurate for the purpose, and a more detailed model is needed. To keep the model simple, assume that $\lambda_t = 0$. Assume that at most one permanent failure in the system may be rectified at a time. Studies of the ARM fault management have shown that, as long as there are other non-failed replicas in a group, a single replica may be started and synchronized in a negative exponentially distributed time with expectation μ_n^{-1} after a replica failure in the group. If all, or the last, replicas in the group fail, the entire group is restarted in a negative exponentially distributed time with expectation μ_{nn}^{-1} after the last replica failure.

g) Give two reasons why a reliability block diagram *can not* be used for analysis in this case. Draw a complete state diagram of the system providing the service, i.e., include all states and transitions, for the case where $n = 3$. Denote a state with $[a, b]$ where a is the number of working hosts in the system and b is the number of working RAS. (Hint: to ease the devising, as well as the interpretation, of the diagram, take care to give it a structure reflecting the operational modes of and the symmetries in the system.) Indicate which states that represent a working and failed service.

A reliability block diagram can not be used since:

- There is a sequential repair of the failed hosts,
- The repair of the replicas are not independent and will depend on the state of the space
- When accurate, we cannot assume independence between permanent and transient failures of the host as well as the server as done in d).

The diagram may be given a structure with a specific number of working hosts in each line and the number of working replicas in each row. Se figure below.



In the following question, we are interested in the reliability of the resource allocation service as seen from a client. Assume that a resource is allocated to a client for identically independently negatively exponentially distributed times with expectation θ^{-1}, i.e., the client refreshes its resource allocation according to a Poisson process with intensity θ. A client experiences a service failure when it, at the expiration of the allocation time, finds the service down. When the regarded service delivery starts, we assume the system is in steady state[2]. The time until the service delivery to the client fails

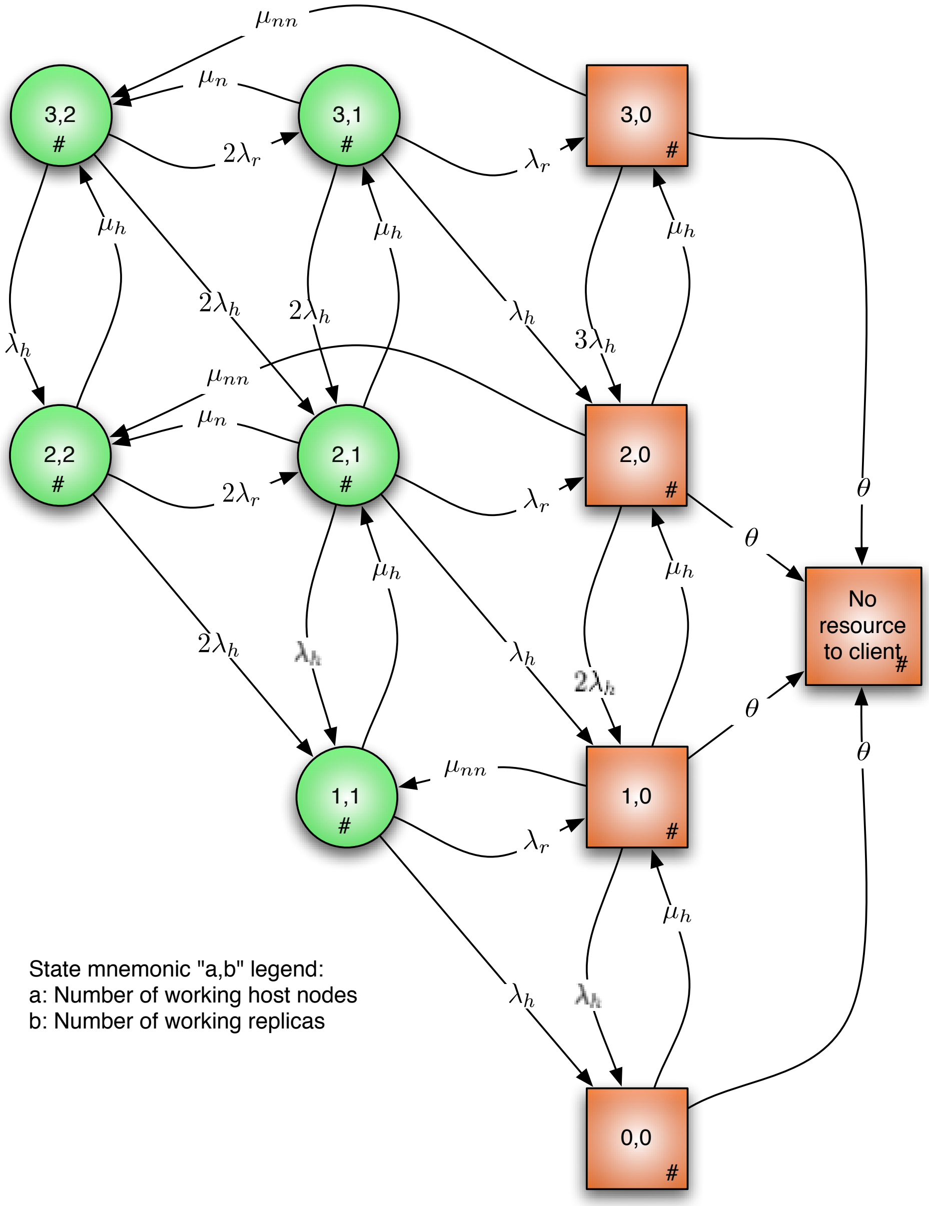
[2] The system is stationary.

is denoted T_F. The reliability function of the system as seen from the client $R_c(t) = \Pr(T_F > t)$ should be obtained.

h) Extend and/or modify the the state diagram found in question g) so it may be used to find $R_c(t)$. (NB! make it clear which part of the diagram that belongs to question g) and which part that belongs to this question.)[3]

The service fails *only* when the client sends a refresh request, and the service is down. Hence, that the service is down does not necessarily incur a service failure. We may regard the down state of the service as sampled by the clients with an intensity θ, which, if it takes place results in a failure of the service provision. This is illustrated in the figure below, where an absorbing state accounting for the client failure is introduced.

[3] If question g) is not solved, use a generic (arbitrary) diagram with up- and down-states and extend this diagram instead.



State mnemonic "a,b" legend:
a: Number of working host nodes
b: Number of working replicas