

Egzamin z systemów rozproszonych

3 lutego 2011

Część: II (piszą osoby niezadowolone z wyniku kolokwium)

Czas pracy: 1.5 h

Punkty możliwe do zdobycia: 25

Każde pytanie na osobnej kartce.

Pytanie 6 (5 pkt). Samo-organizacja (ang. *self-organization*) jest pożądaną właściwością systemów rozproszonych.

- Wymień co najmniej jedną istotną przyczynę powyższego faktu (1 pkt).
- W kilku zdaniach wyjaśnij, jak działa model sprzężenia zwrotnego (ang. *feedback control*) i jak jest on wykorzystywany do budowania samo-organizujących się systemów rozproszonych (3 pkt).
- Podaj i streść w jednym-dwóch zdaniach co najmniej dwa przykłady (z wykładu lub ćwiczeń) samo-organizujących się systemów lub algorytmów (1 pkt).

Pytanie 7 (5 pkt). Wirtualizacja (ang. *virtualization*), zwłaszcza w ostatnich latach, zyskała na popularności.

- Wymień co najmniej dwa powody, dla których używa się wirtualizacji w systemach rozproszonych (1 pkt).
- W kilku zdaniach opisz dwie metody wirtualizacji. Dla każdej metody podaj co najmniej jeden przykład istniejącego systemu/oprogramowania (4 pkt).

Pytanie 8 (10 pkt). Z jednej strony migracja uruchomionego kodu (ang. *code migration*) pomiędzy różnymi maszynami ma potencjalnie wiele zastosowań w systemach rozproszonych. Z drugiej strony, w zależności od formy mobilności, jest ona dość trudna, jako że aktywny kod może mieć wiele zależności, na przykład, może używać różnych lokalnych zasobów.

- W jednym-dwóch zdaniach wyjaśnij na czym polega różnica pomiędzy słabą (ang. *weak mobility*) a silną (ang. *strong mobility*) formą migracji kodu (1 pkt).
- Rozważmy silną formę migracji kodu. Załóżmy, że procesy w naszym systemie używają trzech różnych rodzajów zasobów. Zasoby pierwszego rodzaju, nazwijmy je *unattached*, są łatwo przenaszalne pomiędzy maszynami (np. pojedyncze pliki). Zasoby drugiego rodzaju, nazwijmy je *fastened*, mogą być przeniesione (lub skopiowane) pomiędzy maszynami, ale jest to bardzo drogie (np. duże bazy danych). Zasoby trzeciego rodzaju, *fixed*, nie mogą być przeniesione (np. drukarka lub inny sprzęt).

Proces może być związany z danym zasobem na trzy różne sposoby. Wiązanie przez identyfikator (ang. *binding by identifier*) odpowiada sytuacji, gdy proces używa konkretnej instancji zasobu i odwołuje się do tej instancji poprzez jej globalny identyfikator, na przykład, URL. Wiązanie przez wartość (ang. *binding by value*) odpowiada sytuacji, gdy jedynie wartość a nie instancja zasobu jest ważna (np. istnienie standardowych bibliotek C++ na lokalnej maszynie, ale niekoniecznie w konkretnym katalogu). Wiązanie przez typ (ang. *binding by type*) odpowiada sytuacji, gdy jedynie typ zasobu jest ważny (np. drukarka, monitor).

Przenosząc aktywny kod z jednej maszyny (źródłowej) na drugą (docelową), dla każdego zasobu lokalnego, z którym ten kod jest powiązany, można rozważyć jedną z następujących akcji: (1) *GR* – stworzenie globalnej referencji do tego zasobu i używanie go zdalnie z docelowej maszyny, (2) *MV* – przeniesienie zasobu razem z kodem na docelową maszynę, (3) *CP* – skopiowanie zasobu na docelową maszynę, (4) *RB* – związanie się z lokalnie dostępnym zasobem na docelowej maszynie zamiast z zasobem na maszynie źródłowej. Dla każdej kombinacji rodzaj zasobu-wiązanie przedyskutuj, które z podanych akcji są sensowne i dlaczego oraz która z sensownych akcji jest zwykle preferowana. Innymi słowy, wypełnij poniższą tabelkę i umotywuuj w kilku zdaniach wartości w każdej kolumnie (9 pkt).

	unattached	fastened	fixed
by identifier	GR?, MV?, CP?, RB?		
by value			
by type			

Pytanie 9 (5 pkt). Plotkowanie (ang. *gossiping*) to efektywna forma propagowania informacji, nie tylko w systemach rozproszonych.

- Wyjaśnij w kilku zdaniach na czym polegają strategie *push*, *pull* i *push-pull* plotkowania. Wymień wady i zalety każdej strategii w zależności od tego w jakim stopniu dana porcja informacji jest rozpropagowana w systemie (3 pkt).
- Plotkowanie zwykle wymaga wybierania losowych węzłów z sieci (ang. *peer sampling*). Streść w paru zdaniach, jak można dostarczyć tego mechanizmu używając samego plotkowania (1.5 pkt).
- W jednym-dwóch zdaniach streść praktyczną wadę tego podejścia do wybierania węzłów (0.5 pkt).

Distributed Systems: Final Exam**February 3, 2011****Part: II (only the people dissatisfied with their midterm results should write)****Exam duration: 1.5 h****Total points: 25****Each question should be answered on a separate sheet.****Question 6 (5 pts).** Self-organization capabilities are desirable in distributed systems.

- Give at least one reason for the above statement (1 pt).
- In a few sentences, explain the feedback control loop and the way it can be used to build self-organized distributed systems (3 pts).
- Give and summarize in one-two sentences at least two examples of self-organized systems or algorithms (mentioned during the lectures or student presentations) (1 pt).

Question 7 (5 pts). Virtualization has been gaining popularity.

- Give at least two reasons for using virtualization in distributed systems (1 pt).
- In a few sentences, describe two methods for virtualization. For each of these methods, give at least one example of an existing system/software (4 pts).

Question 8 (10 pts). On the one hand, code migration between machines has many potential applications in distributed systems. On the other hand, depending on how precisely the code is migrated, migration can be challenging, especially since an active piece of code can have many run-time dependencies. For example, it can use different local resources.

- In one-two sentences, explain the difference between weak and strong code mobility (1 pt).
- Consider strong code mobility. Let us assume that processes in our system use three different types of resources. First, *unattached* resources can be migrated easily between machines (e.g., individual files). Second, *fastened* resources can be migrated (or copied) from one machine to another, but only at a high cost (e.g., large databases). Third, *fixed* resources cannot be migrated (e.g., printers or other hardware).

A process can be bound to a resource in three different ways. First, a process bound to a resource *by identifier* uses a particular instance of the resource and refers to that instance by the instance's global identifier, for example, a URL. Second, a process is bound to a resource *by value* if the value and not a particular instance of the resource are important for the process, for example, standard C++ libraries exist on the machine, but not necessarily in a particular directory. Third, a process is bound to a resource *by type* if only the type of the resource is relevant for the process, for example, any printer or monitor can do.

Consider a situation in which some code is being moved from a source machine to a destination machine. For each local resource to which the code is bound, one of the following actions can be taken: (1) *GR* – establishing a global reference to this resource and using the resource remotely from the destination machine, (2) *MV* – moving the resource together with the code to the destination machine, (3) *CP* – copying the resource to the destination machine, and (4) *RB* – re-binding to a resource locally available on the destination machine instead of the current resource on the source machine. For each resource type-binding combination, discuss which of the above actions are sensible and why, and which of those actions should normally be preferred. In other words, fill in the table below and motivate in a few sentences the values in each cell (9 pts).

	unattached	fastened	fixed
by identifier	GR?, MV?, CP?, RB?		
by value			
by type			

Question 9 (5 pts). Gossiping is an efficient method for propagating information, not only in distributed systems.

- In a few sentences, explain the following gossiping strategies: push, pull, and push-pull. Enumerate advantages and disadvantages of each of these strategies depending on the degree to which a piece of information has been propagated in the system (3 pts).
- Gossiping requires random node sampling. Summarize in a few sentences how random peer sampling can be implemented using gossiping itself (1.5 pts).
- In one-two sentences, describe a significant practical drawback of this approach to peer sampling (0.5 pts).