Robert R. CHODOREK, Agnieszka CHODOREK

WIRTUALIZACJA SYSTEMÓW KOMPUTEROWYCH JAKO ŚRODEK ZAPEWNIENIA NIEZAWODNOŚCI

Niniejszy artykuł poświęcony jest dyskusji nad metodami wirtualizacji sprzętu komputerowego i jej zastosowań do zapewnienia zwiększonej niezawodności systemu sprzętowo-programowego. Systemy takie mogą być stosowane obok (a w niektórych przypadkach: zamiast) tradycyjnego powielania wybranych elementów infrastruktury krytycznej. W artykule omówiono zagadnienia związane z wirtualizacją systemów komputerowych. Przedstawiono podstawowe typy wirtualizacji, jej zastosowanie i stosowane w praktyce rozwiązania. W artykule zaprezentowano również wyniki badań systemu wirtualizacji pod kątem jego wydajności, zorientowane na realizację konkretnego mechanizmu rezerw (konkretnej struktury niezawodnościowej).

WSTĘP

Systemy wieloprocesorowe, wielordzeniowe umożliwiają współbieżne wykonywanie wielu procesów. Niestety, większość dostępnego oprogramowania nie jest w stanie wykorzystać wielu jednostek wykonawczych lub wykorzystuje je w stopniu niewystarczającym w stosunku do możliwości konkretnego sprzętu. Z tego powodu, zamiast wykorzystywać wiele procesorów (i/lub wiele rdzeni) do uruchomienia pojedynczej aplikacji, stosowane są rozwiązania odwrotne - umożliwiające wykorzystanie danego, fizycznego sprzętu komputerowego przez wiele aplikacji lub przez wiele systemów operacyjnych. Takim rozwiązaniem jest wirtualizacja.

Problemy związane z niepełnym wykorzystaniem wydajnego systemu komputerowego pojawiły się bardzo wcześnie, bo już pod koniec lat pięćdziesiątych dwudziestego wieku. Pomimo, iż wirtualizacja jest zwykle postrzegana jako nowa, obiecująca gałąź informatyki, pierwsze rozwiązanie wirtualizacji zostało wprowadzone pół wieku temu [1]. W latach 60. ubiegłego wieku, w systemach firmy IBM, w ramach projektu M44/44X wprowadzono fizyczny system komputerowy M44, na którym uruchamiano maszyny wirtualne 44X. Rozwiązanie to miało na celu zwiększenie efektywności wykorzystania wydajnego systemu komputerowego M44, na którym, w środowisku maszyny wirtualnej 44X, uruchamiane były procesy użytkowników. Aby nie trzeba było zmuszać użytkownika do modyfikacji oprogramowania, utworzone środowisko wirtualne było niejako kopią oryginalnego, fizycznego systemu komputerowego. Z punktu widzenia użytkownika, każdy z systemów wirtualnych był niejako kopią oryginalnego, fizycznego systemu komputerowego.

Wirtualizacja systemów komputerowych jest nie tylko sposobem na lepsze wykorzystanie sprzętu, ale również techniką umożliwiającą abstrakcję zasobów systemu komputerowego. Tego typu abstrakcja umożliwia uniezależnienie się użytkownika od sprzętu, na którym jest uruchamiane oprogramowanie. Wirtualizacja umożliwia bezpieczne testowanie nieznanego lub niepewnego oprogramowania. Dostanie ono dostęp jedynie do wydzielonych zasobów, pozostających pod pełną kontrolą maszyny wirtualnej. Dzięki temu (o ile nie nastąpi celowy atak na maszynę wirtualną, a jedynie awaria lub atak na system operacyjny) praktycznie nie ma niebezpieczeństwa zniszczenia fizycznych zasobów danego komputera, a tylko zamkniętego środowiska zwirtualizowanego.

Odrębnym zagadnieniem jest niezawodność systemu komputerowego. Zastosowanie urządzeń redundantnych w celu poprawy niezawodności dotyczy głównie serwerów i, w najprostszym przypadku, związane jest z rezerwą sprzętową urządzeń podatnych na uszkodzenia mechaniczne - przede wszystkim dysków i zasilaczy. Wobec pozostałych elementów przyjmuje się założenie, że są one wystarczająco niezawodne. Jeżeli koszt awarii systemu komputerowego jest duży, takie założenie jest zbyt optymistyczne. Można wtedy zwiększyć niezawodność systemu stosując kombinację zwielokrotnienia maszyn i technik wirtualizacji.

W zależności od zastosowanego rozwiązania technicznego, w środowisku wirtualnym mogą być uruchamiane: (1) pojedyncze programy oraz (2) całe systemy operacyjne. Pierwszy przypadek jest to tzw. maszyna wirtualna poziomu użytkownika (ang. *user level virtual machine, user level VM*). Określenie to dotyczy uruchamiania aplikacji w ramach systemu operacyjnego lub specjalizowanego środowiska uruchomieniowego, np. maszyny wirtualnej Java. Aplikacja korzysta wówczas z udostępnianego interfejsu ABI (ang. *application binary interface*), który definiuje, jakie instrukcje procesora są dostępne dla uruchomionej aplikacji i jakie są dostępne dla niej urządzenia wejścia-wyjścia.

Drugi przypadek to tzw. maszyna wirtualna poziomu systemu operacyjnego (ang. *system level VM*). Termin ten dotyczy uruchamiania wirtualizowanego systemu operacyjnego gościa (ang. *guest*) w środowisku systemu operacyjnego gospodarza (ang. *host*) lub środowiska wirtualizacyjnego (ang. *hypervisor*).

Niniejszy artykuł poświęcony jest tematyce zwiększania niezawodności systemu komputerowego z wykorzystaniem maszyn wirtualnych poziomu systemu operacyjnego. Artykuł składa się z trzech rozdziałów. W rozdziale pierwszym omówiono zagadnienia wirtualizacji i opisano typowe środowiska umożliwiające realizację wirtualizacji poziomu systemu operacyjnego. W rozdziale drugim przedyskutowane zostały aspekty związane z zapewnieniem niezawodności, a rozdział trzeci analizuje wydajność systemu wirtualizacji pod kątem zapewnienia niezawodności.

1. WIRTUALIZACJA I TECHNIKI WIRTUALIZACYJNE

Rozdział omawia zagadnienia zarządzania systemem wirtualnym oraz związku wykorzystywanego poziomu uprzywilejowania z techniką wirtualizacji

1.1. Zarządzanie

Wirtualizacja pozwala na abstrakcyjne postrzeganie systemu komputerowego. W praktyce oznacza to, że stosowane techniki umożliwiają utworzenie pełnego wirtualnego środowiska, które przez system operacyjny gościa jest postrzegane jako fizyczny system komputerowy o określonych zasobach sprzętowych, tj. procesorze, pamięci o określonej pojemności, określonych kartach urządzeń wejścia-wyjścia (jak karty grafiki, karty sieciowe czy muzyczne). Tym samym, wirtualizacja pozwala na rozseparowanie dwu warstw systemu komputerowego, warstwy systemu operacyjnego i warstwy sprzętu, sprzętem i oprogramowaniem maszyny wirtualnej.

Przydział zasobów sprzętowych systemowi operacyjnemu oraz komunikowanie się rozseparowanych warstw systemu operacyjnego i sprzętu zapewniają odpowiednie techniki wirtualizacji. Techniki te zapewniają również zarządzanie sprzętowymi zasobami komputera. Do zarządzania wirtualizacją (a konkretniej, maszynami wirtualnymi) służy hiperwizor[1] (ang. *hypervisor*, tłum. również jako hipernadzorca). Jest to system kontrolno-zarządzający czasu rzeczywistego, którego zadania i stopień skomplikowania bywają porównywane do uproszczonego systemu operacyjnego[2].

R. P. Goldberg, w swojej fundamentalnej pracy [3] z roku 1974, wyróżnia dwa typy hiperwizorów: hiperwizor typu 1 (ang. *Type-1 hipervisor*[3]) oraz hiperwizor typu 2 (ang. *Type-2 hipervisor*[4]). Hiperwizor typu 1 pracuje bezpośrednio na sprzęcie komputerowym. Jest to odrębny system zarządzający, niezależny od systemu operacyjnego gospodarza. Musi zatem zawierać zestaw sterowników do urządzeń systemu komputerowego oraz własny program szeregujący (ang. *scheduler*) do zarządzania procesami.

Hiperwizor typu 2 nie jest posadowiony na "gołym" sprzęcie lecz na systemie operacyjnym gospodarza, uruchomionym na sprzęcie. Jest to wirtualizacja z poziomu systemu operacyjnego, gdzie system operacyjny gospodarza obsługuje kilka odseparowanych od siebie środowisk wirtualnych w ramach tego samego systemu. Hiperwizor korzysta z istniejących sterowników systemu hosta. Należy jednak pamiętać, że system operacyjny gospodarza musi być bardzo stabilny, gdyż niestabilność, błędy i inne nieprawidłowości systemu operacyjnego przenoszą się na wszystkie maszyny wirtualne uruchomione na bazie tego systemu.

Typ hiperwizora odpowiada klasyfikacji wirtualizacji na wirtualizację typu pierwszego i typu drugiego. Przykładem wirtualizacji typu pierwszego jest Microsoft Hyper-V [2]. Przykładem wirtualizacji typu drugiego są maszyny wirtualne emulatora Netkit [5].

Powyższa klasyfikacja, wprowadzona przez Goldberga, była wielokrotnie powielana lub odkrywana na nowo (czasem z drobnymi różnicami), przez wielu autorów - stąd, przypuszczalnie, wzięła się mnogość określeń. W praktyce klasyfikacja ta nieco się zaciera, przynajmniej jeśli chodzi o konkretne implementacje. Bywają implementacje, np. linuksowe, które potrafią pracować w trybie 2-gim, ale jednocześnie modyfikują jądro Linuxa tak, jakby Linux (tu: system operacyjny gospodarza) był hiperwizorem typu 1.

W pracy [1] można znaleźć klasę hiperwizorów pośrednią w stosunku do klasyfikacji Goldberga, tj. hybrydowy VMM. Łączy on w sobie oba goldbergowskie rozwiązania. Jądro VMM samodzielnie kontroluje przydział procesora, ale do obsługi urządzeń wykorzystuje sterowniki pobierane z systemu operacyjnego gospodarza.

1.2. Poziomy uprzywilejowania

Zarządzanie wirtualizacją wiąże się z koniecznością przydzielenia odpowiednich priorytetów hiperwizorowi, systemowi gościa i systemowi procesora. W przypadku typowych systemów operacyjnych wymagane są przynajmniej dwa poziomy uprzywilejowania[5] procesora – jeden dla użytkownika (niższy poziom uprzywilejowania), drugi dla systemu operacyjnego (wyższy poziom uprzywilejowania). Powszechnie stosowane procesory rodziny x86, produkowane przez firmy Intel i AMD, mają 4 poziomy uprzywilejowania. Najwyższym poziomem uprzywilejowania jest poziom 0, a najniższym poziom 3. Typowo systemy komputerowe korzystają tylko z dwóch poziomów. Poziomu 0, na którym funkcjonuje system operacyjny i poziomu 3, na którym są uruchamiane procesy użytkowników. Poziomy 1 i 2 zazwyczaj nie są wykorzystywane.

Ze względu na wykorzystywany poziom uprzywilejowania, techniki wirtualizacji można podzielić na:

- parawirtualizację (ang. *partial virtualization*, *paravirtualization*),
- pełną wirtualizację (ang. full virtualization),
- wirtualizację wspomaganą sprzętowo (ang. har*dware-assisted virtualisation*).

Parawirtualizacja wiąże się z modyfikacją jądra systemu operacyjnego gościa. Realizacja parawirtualiacji wymaga bowiem przeniesienia wirtualizowanego systemu operacyjnego z poziomu 0 do poziomu 1. Umożliwi to, m.in., wywłaszczanie s.o. zgodnie z odpowiednim szeregowaniem, a tym samym poprawne działanie modułu zarządzającego maszynami wirtualnymi. Niestety wymaga to pełnego dostępu do kodu źródłowego systemu operacyjnego i zmiany tych fragmentów, które są uzależnione od wykorzystywanego poziomu ochrony procesora.

W przypadku pełnej wirtualizacji, system operacyjny gościa jest niezmieniony. Pracuje on jednak na 3 poziomie uprzywilejowania, jak każda inna aplikacja systemu hosta. Aby możliwe było wykonywanie instrukcji uprzywilejowanych zwirtualizowanego systemu gościa, wirtualizacja jest realizowana za pomocą translacji binarnej kodu (translacji podlegają instrukcje uprzywilejowane).

Wirtualizacja wspomagana sprzętowo również nie wymaga zmian w kodzie systemu operacyjnego gościa. Wykonywanie instrukcji uprzywilejowanych systemu gościa odbywa się za pomocą wspomagania sprzętowego, często ze sprzętowego wsparcia ze strony procesorów. Sprzętowe wspomaganie wirtualizacji dla platformy x86 jest oferowane zarówno przez firmę Intel (VT-x, VT-d, VT-d2, VT-c) jak i AMD (AMD-V, ang. *AMD virtualization*). Rozwiązanie VT-x firmy Intel dostarcza nowego poziomu uprzywilejowania „*root* VMX" – o wyższym poziomie uprawnień niż poziom 0 (stąd czasem nazwa poziom -1). Rozwiązanie AMD-V firmy AMD działa podobnie, różnice dotyczą szczegółów implementacyjnych. Sprzętowe wspomaganie funkcji wejścia-wyjścia (w przypadku firmy Intel posiadającego oznaczenia VT-d i VT-d2) jest techniką wspomagającą wirtualizację w zakresie nadzoru nad urządzeniami wejścia-wyjścia. Technika VT-c stanowi wspomaganie wirtualizacji transmisji. Rozwiązania VT-d, VT-d2 i VT-c typowo wymaga uruchomienia mechanizmów VT-x.

---

[1] Spotykane są również nazwy: monitor maszyn wirtualnych (ang. *Virtual Machine Monitor*, *VMM*) oraz, podobna, zarządca (menadżer) maszyn wirtualnych (ang. *virtual machine manager*, również *VMM*). Uwaga: VMM to także nazwa wewnętrznego rozwiązania dystrybucji Linuxa Redhat; rozwiązanie to jest wykorzystywane do zarządzania różnymi hiperwizorami.

[2] W literaturze anglojęzycznej hiperwizor określany bywa czasami jako "*light-weight operating system*" - dosł. "lekki", "odchudzony" system operacyjny.

[3] W języku angielskim spotykane są również dwie nazwy idiomatyczne: *native hypervisor* (spolszczana czasami do: natywny hiperwizor) oraz *bare-metal hypervisor* (nieprzetłumaczalna). Odpowiednikiem terminu *native* jest m.in. "występujący naturalnie", "rodzimy". "*Bare metal*" to po polsku dosłownie "goły metal" - niemalowany lub (w przypadku przewodu) nieizolowany, bez izolacji. *Native hypervisor* odnosi się do naturalnego, intuicyjnego sposobu nadzoru prowadzonego przez hiperwizora typu 1. *Bare-metal hypervisor* to hiperwizor bezpośrednio posadowiony na sprzęcie, przen. "dotykający gołego hardware'u", "dotykający gołego metalu, gołego kabla".

[4] W j. ang. spotykana jest również nazwa "*hosted hypervisor*", dosł. "hiperwizor (u)goszczony" (w domyśle: przez system operacyjny).

[5] Równoznaczne z poziomami ochrony.

Pełna wirtualizacja i wirtualizacja wspomagana sprzętowo dwiema najczęściej stosowanymi technikami wirtualizacji. W zależności od zastosowanych rozwiązań, techniki ta mogą posiadać różną wydajność na tym samym sprzęcie. Podobnie, jak miało to miejsce w przypadku wirtualizacji typu pierwszego i drugiego, tu również spotykane są praktyczne implementacje, posługujące się techniką mieszaną.

Osobną kategorią wirtualizacji, nie ujętą w powyższe ramy, jest emulacja (ang. *emulation*). Pojęcie to w odniesieniu do technik wirtualizacyjnych oznacza emulowanie innego sprzętu niż ten, który jest w dyspozycji. W tym miejscu należy wspomnieć, że elementy emulacji mogą wystąpić w każdej z powyższych technik wirtualizcyjnych. Zwykle jednak jest to emulacja cząstkowa, dotycząca pojedynczych urządzeń, z pominięciem procesora. Typowe techniki wirtualizacyjne dają procesor taki, jaki został zainstalowany, co najwyżej może być on trochę zubożony. W przypadku emulacji, w środowisku zwirtualizowanym emulowane są wszystkie elementy systemu komputerowego, łącznie z typem procesora. Dzięki temu, na fizycznym systemie komputerowym, możliwe jest uruchamiane programów lub systemów operacyjnych przeznaczonych dla innej platformy sprzętowej lub sprzętowo-programowej (np. emulacja architektury PowerPC na komputerze z procesorem x86).

Tab. 1. *Przykładowe rozwiązania technik wirtualizacyjnych*

Rozwiązanie	Technika wirtualizacji
VMware ESX Server	pełna wirtualizacja
VMware Server/Workstation	pełna wirtualizacja
Xen	pełna wirtualizacja lub parawirtualizacja
KVM	pełna wirtualizacja
Microsoft Hyper-V	pełna wirtualizacja
QEMU	emulacja
UML	parawirtualizacja
VirtualBox	pełna wirtualizacja

Obecnie stosowanych jest kilka rozwiązań wirtualizacji, z których najpopularniejsze zostały zamieszczone w tabeli 1.

Wirtualizacja jest środkiem wspomagającym wykorzystanie systemu operacyjnego, poprawiającym jego bezpieczeństwo i niezawodność. Tym niemniej trzeba pamiętać, że spośród rozpatrywany wyżej technik realizacji środowiska wirtualnego, emulacja i parawirtualizacja nie spełniają kryteriów Popka i Goldberga skuteczności maszyny wirtualnej[6]. Także część rozwiązań zrealizowanych w dwóch pozostałych technikach (wirtualizacja wspomagana sprzętowo i pełna wirtualizacja) może nie spełniać tych kryteriów.

2. WIRTUALIZACJA JAKO WSPARCIE TECHNICZNE DLA SYSTEMU ZAPEWNIENIA NIEZAWODNOŚCI - STUDIUM PRZYPADKU

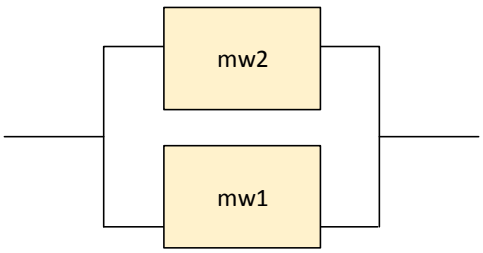
Ogólnie rzecz ujmując, niezawodność obiektów technicznych można zwiększać na dwa sposoby. Pierwszy to wymiany profilaktyczne, typowo stosowane do zwiększania niezawodności urządzeń mechanicznych. Drugi polega na rezerwowaniu (redundancji). Sposób ten stosuje się do poprawy niezawodności urządzeń i układów elektroniki, telekomunikacji, informatyki, w tym również do zwiększania niezawodności systemów komputerowych.

Systemy logistyczne intensywnie wykorzystują narzędzia informatyczne, w tym bazy danych wraz z systemami zarządzania bazami. Wymagają przy tym (podobnie, jak systemy telekomunikacyjne czy bankowe), by takie narzędzie było możliwie jak najbardziej niezawodne. Oznacza to, między innymi, że powinno charakteryzować się możliwie jak największą dostępnością (ang. *availability*).

Systemy informatyczne o wysokim poziomie dostępności (ang. *high availability*, HA) są to systemy odporne na uszkodzenia. Aby system był systemem HA, musi nie tylko charakteryzować się odpowiednimi wartościami wskaźników niezawodności (jak średni czas pomiędzy uszkodzeniami). Musi również w sposób automatyczny, w czasie rzeczywistym, radzić sobie z pojedynczymi awariami.

Jest to możliwe dzięki zastosowaniu redundancji. Sama redundancja, w przypadku serwerów, nie jest niczym nowym. Od dawna można spotkać (nawet w najniższym segmencie cenowym) urządzenia z podwojonymi zasilaczami, powielonymi wentylatorami, czy z macierzami dyskowymi RAID. Rodzajem redundancji są również kopie zapasowe, choć czas odtwarzania stanu systemu z kopii jest zbyt duży, a samo odtwarzanie zwykle zbyt niepełne, by mogły one pełnić rolę rezerwy dla systemu o dużej dostępności. Pomijając kosztowne rozwiązania sprzętowe, taka rezerwa staje się możliwa m.in. dzięki wprowadzeniu wirtualizacji.

Uruchomienie systemu operacyjnego w maszynie wirtualnej oferuje dodatkowe możliwości funkcjonalne. Możliwe jest, między innymi, zapisanie stanu systemu operacyjnego maszyny wirtualnej (ang. *snapshot*), dzięki czemu można odtworzyć ten stan na żądanie i wrócić do zapisanego momentu wykonywania. Dzięki uruchamianemu na żądanie mechanizmowi zapisu i odczytu możliwa stała się migracja (ang. *migration*) maszyny wirtualnej z jednego systemu fizycznego do innego systemu. Podczas migracji, w aktualnym systemie fizycznym zapisywany jest stan bieżący maszyny i maszyna jest zatrzymywana. W docelowym systemie, bieżący stan maszyny jest odtwarzany i maszyna jest uruchamiana począwszy od tego stanu. Podobnym rozwiązaniem jest przeniesienie maszyny wirtualnej na inny system w przypadku awarii (ang. *failover*). Mechanizm ten jest podstawą tworzenia systemów rezerwowania.



Rys. 1. *Równoległa struktura niezawodnościowa.*
Legenda: mw1 - maszyna wirtualna będąca elementem podstawowym, mw2 - maszyna wirtualna będąca elementem rezerwowym.

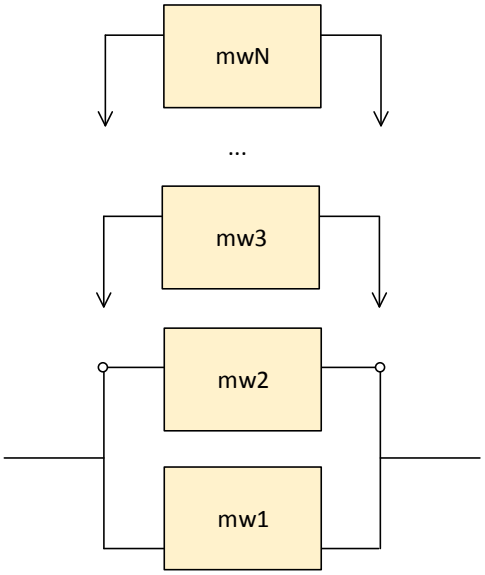
W przypadku rezerwy gorącej, nazywanej również rezerwą obciążoną, w obiekcie technicznym oprócz elementu podstawowego dodaje się jeden lub więcej elementów rezerwowych. Nazwa rezerwy pochodzi od tego, że zarówno element podstawowy, jak i element(y) rezerwowy(e) są normalnie eksploatowane (obciążone bieżącymi zadaniami) i tworzą strukturę równoległą. Obiekt techniczny o strukturze równoległej jest zdatny (sprawny), jeżeli co najmniej jeden z jego elementów składowych jest zdatny. Schemat blokowy takiej struktury, narysowany dla najprostszego przypadku (rezerwy gorącej z jednym elementem podstawowym i jednym rezerwowym), pokazany został na rysunku 1.

Zwiększenie niezawodności systemu informatycznego z wykorzystaniem maszyn wirtualnych opiera się na założeniu, że hiperwizory systemów komputerowych połączone są przez sieć kompute-

[6] Kryteria skuteczności maszyny wirtualnej, sformułowane przez Popka i Goldberga [4] to: (1) odpowiedniość (równoważność) (ang. *Equivalence*/*Fidelity*) - program uruchomiony w środowisku wirtualnym ma zachowywać się tak samo jak zachowywałby się, gdyby był uruchomiony na fizycznym sprzęcie, (2) kontrola zasobów (ang. *Resource controls/Safety*) – maszyna wirtualna musi mieć pełną kontrolę nad zasobami, które są udostępniane w wirtualizowanym środowisku, (3) wydajność (ang. *Efficiency*/*Performance*) – znacząca część instrukcji programu uruchamianego w środowisku wirtualnym musi być wykonywana bez ingerencji maszyny wirtualnej w wykonywany kod.

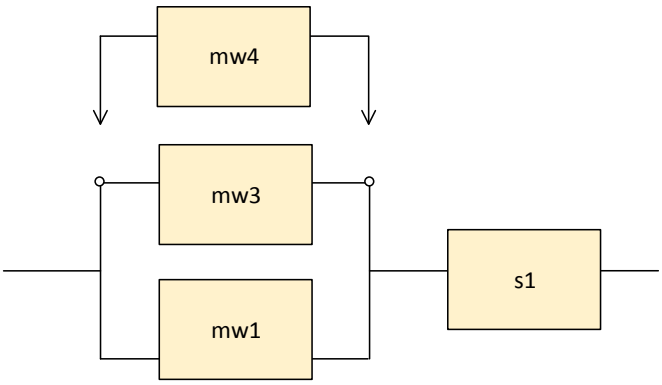
rową, tworząc klaster. Rysunek 1 obrazuje typową sytuację, w której na dwóch niezależnych systemach fizycznych uruchamiane są identyczne maszyny wirtualne (mw1 i mw2). Maszyny te pracują równolegle, synchronizując ciągle zawartość swoich dysków i stanu systemu. Aby uniknąć problemów z integralnością baz danych, system zarządzający wskazuje, z którą maszyną w danym momencie współpracują użytkownicy. W przypadku wykrycia awarii użytkownicy zostaną przekierowani na sprawny system, który będzie kontynuował rozpoczętą operację.

Tego typu rozwiązanie może być postrzegane jako rezerwa ciepła (częściowo obciążona), gdyż użytkownicy obciążają tylko jedną z maszyn wirtualnych równoległej struktury niezawodnościowej. Druga zaś, można uznać, iż pracuje jałowo. Jednak, aby uzyskać wysoki poziom dostępności, drugi system musi znajdować się w takim samym stanie jak pierwszy (z dokładnością do czasów sygnalizacji), wykonując te same operacje. Zatem komponenty systemu komputerowego są obciążane identycznie w obydwu systemach, łącznie z podsystemem komunikacyjnym. Autorzy zatem skłaniają się ku stwierdzeniu, że jest to rezerwa gorąca.



Rys. 2. *Mieszana (tu: równoległo-przekaźnikowa) struktura niezawodnościowa klastra.*

Jeżeli w klastrze utworzonym dzięki połączeniu hiperwizorów siecią komputerową są dostępne moce przerobowe (co w praktyce oznacza, że są w nim fizyczne systemy komputerowe o wystarczających zasobach), to po uszkodzeniu bezzwłocznie tworzona jest kopia pracującego systemu rezerwowego. W ten sposób powstaje kolejna rezerwa, która dołączana jest do pracującej maszyny, tworząc wraz z nią równoległą strukturę niezawodnościową. O ile zasoby klastra na to pozwolą, równoległa struktura niezawodnościowa będzie odtwarzana po każdej awarii. Maszyna wirtualna, która albo sama uległa uszkodzeniu, albo pracowała na uszkodzonym serwerze, jest bezpowrotnie tracona. Maszyna wirtualna jest elementem nienaprawialnym, mimo, iż system komputerowy, na którym została posadowiona, jest naprawialny.



Rys. 3. *Mieszana struktura niezawodnościowa klastra, zbudowana z połączenia trzech struktur podstawowych.*

Analizowany klaster nie tworzy zatem "czystej" struktury równoległej lecz strukturę mieszaną, w której oprócz struktury równoległej, tworzącą rezerwę gorącą, występuje również struktura przekaźnikowa (rys. 2), tworząca rezerwę zimną[7] (nieobciążoną). Maszyny te fizycznie nie istnieją - choć, oczywiście, istnieje zarówno sprzęt, na którym zostaną posadowione, jak i uruchomione na nim oprogramowanie hiperwizora. Instancja maszyny wirtualnej wchodzącej w skład rezerwy zimnej zostanie powołana dopiero w chwili, gdy zajdzie potrzeba zastąpienia uszkodzonej maszyny rezerwy gorącej.

Zgodnie z zasadą działania systemu z rezerwą zimną, w chwili uszkodzenia elementu podstawowego, jeden z elementów rezerwowych przejmuje jego zadania, przechodząc ze stanu nieobciążenia do stanu obciążenia (stanu normalnej pracy). Kolejne uszkodzenia spowodują załączanie następnych elementów rezerwowych, aż do wyczerpania rezerwy.

W omawianym przykładzie środowiska wirtualnego, odtworzenie stanu po pojedynczym uszkodzeniu odbywa się dwuetapowo. Najpierw odtwarzany jest stan systemu informatycznego, a później, w miarę możliwości, odtwarzany jest stan struktury równoległej. Struktura równoległa zapewnia rezerwowanie dla systemu HA, który po uszkodzeniu maszyny odtwarza stan systemu informatycznego na tyle szybko, że jest to niezauważalne (w najgorszym razie: prawie niezauważalne) dla użytkownika. Struktura przekaźnikowa służy wydłużeniu średniego czasu poprawnej pracy systemu HA.

Na rysunku 3 pokazany został schemat blokowy systemu opartego na klastrze złożonym z czterech hiperwizorów, każdy uruchomiony na innej fizycznej maszynie. Rysunek obrazuje hipotetyczną sytuację, gdy maszyna mw2 z rysunku 2 uległa awarii. Maszyna wirtualna mw3 zastąpiła mw2, przechodząc z rezerwy zimnej do rezerwy gorącej. W rezerwie zimnej pozostała jeszcze mw4. Tego typu system HA będzie zdatny, jeśli wszystkie cztery systemy będą zdatne oraz zdatna będzie sieć komputerowa, w założeniach zapewniająca zarówno łączność z elementami systemu informatycznego (rozproszonymi po klastrze), jak i łączność ze światem zewnętrznym (w tym z użytkownikami systemu informatycznego). System informatyczny i sieć komputerowa tworzą razem szeregową strukturę niezawodnościową[8] (rys. 3).

Należy podkreślić, że o ile w rezerwie zimnej element podstawowy jest wyraźnie wyróżniony jako obciążony, to istotą rezerwy gorącej jest to, że wszystkie elementy (podstawowy i rezerwowe) są obciążone. W obu zaś przypadkach, element rezerwowy jest w stanie pełnić funkcję elementu podstawowego (w rezerwie gorącej: niekiedy również obu elementów). Jeśli maszyny wirtualne zostaną

[7] W systemie z rezerwą zimną, obiekt techniczny jest zdatny, jeżeli co najmniej jeden z jego elementów składowych jest zdatny.
[8] Obiekt techniczny o strukturze szeregowej jest zdatny, jeżeli wszystkie jego elementy składowe są zdatne.

uruchomione na podobnej klasy serwerach, różnica między elementem podstawowym a rezerwowym będzie miała charakter formalny. Maszyną podstawową będzie ta spośród maszyn wirtualnych, do której, w danym momencie, switch wirtualny będzie kierował ruch generowany przez użytkowników. Będzie to zależało od przyjętej polityki przydziału zasobów, a przypisane role elementu podstawowego i zapasowego mogą się zmieniać wraz upływem czasu.

PODSUMOWANIE

Obserwowane w ostatnich latach tendencje do wirtualizacji zasobów sprzętowych dały impuls do nowego spojrzenia na zagadnienia niezawodności systemu komputerowego. Wzrost niezawodności sprzętu komputerowego, wzrost jego wydajności oraz wspomniana już wirtualizacja sprawiły, że typowe metody zwiększania niezawodności, polegające na zastosowaniu rezerwy zimnej lub gorącej (dotychczas występujące w postaci, na przykład, podwojonych lub potrojonych procesorów) stały się bardziej dostępne dla użytkownika. Zaś wzrost wydajności i niezawodności sieci komputerowych spowodował, że możliwe stało się szybkie przenoszenie zasobów na sąsiednie maszyny.

BIBLIOGRAFIA

1. Stasiak A., Skowroński Z., *Wirtualizacja - kierunek rozwoju platform n-procesorowych, Przegląd telekomunikacyjny*, vol. LXXXI, nr 6, 2008.
2. Serafin M., *Wirtualizacja w praktyce. Helion,* Gliwice 2012.
3. Goldberg R. P., *Survey of virtual machine research, Computer*, 7 (6) 1974.
4. Popek G, J., Goldberg R. P., *Formal Requirements for Virtualizable Third Generation Architectures, Communications of the ACM*, Vol. 17, Nr 7, 1974.
5. Chodorek A., Chodorek R., *Możliwości zastosowania emulatora Netkit w badaniach naukowych i dydaktyce, Logistyka* 2014, nr 6.

The virtualization of computer systems as method to ensure reliability

This article is devoted to the discussion of the methods of virtualization the hardware and its applications to ensure increased reliability of the hardware and software system. Such systems can be used alongside (and in some cases: instead of) traditional duplication of selected elements of critical infrastructure. The article discusses the issues associated with virtualization systems. The article shows the basic types of virtualization, its use and actual practices. The article presents the results of the research system virtualization for its performance-oriented implementation of specific provisioning mechanism (specific reliability structure).

Autorzy:
dr inż. **Agnieszka Chodorek** – Politechnika Świętokrzyska, Wydział Elektrotechniki, Automatyki i Informatyki, Katedra Systemów Informatycznych; 25-314 Kielce; al. Tysiąclecia Państwa Polskiego 7. E-mail: a.chodorek@tu.kielce.pl
dr inż. **Robert Chodorek** – AGH Akademia Górniczo-Hutnicza, Wydział Informatyki, Elektroniki i Telekomunikacji, Katedra Telekomunikacji; 30-059 Kraków; Al. A. Mickiewicza 30. E-mail: chodorek@agh.edu.pl