

Wirtualizacja

- Sara Wilczyńska
- Magdalena Zakrzewska
- Aleksander Lewandowski

Definicje

Moim zdaniem najfajniejsza:

Wirtualizacja jest szerokim pojęciem odnoszącym się do abstrakcji zasobów w różnych aspektach komputeryzacji.

bardziej praktyczna:

Wirtualizacja to zestaw metod umożliwiających podział zasobów komputera między wiele środowisk

Podział

W literaturze (sieci :) możemy spotkać się z różnymi podziałami metod wirtualizacji. Najpopularniejszy z nich wyszczególnia:

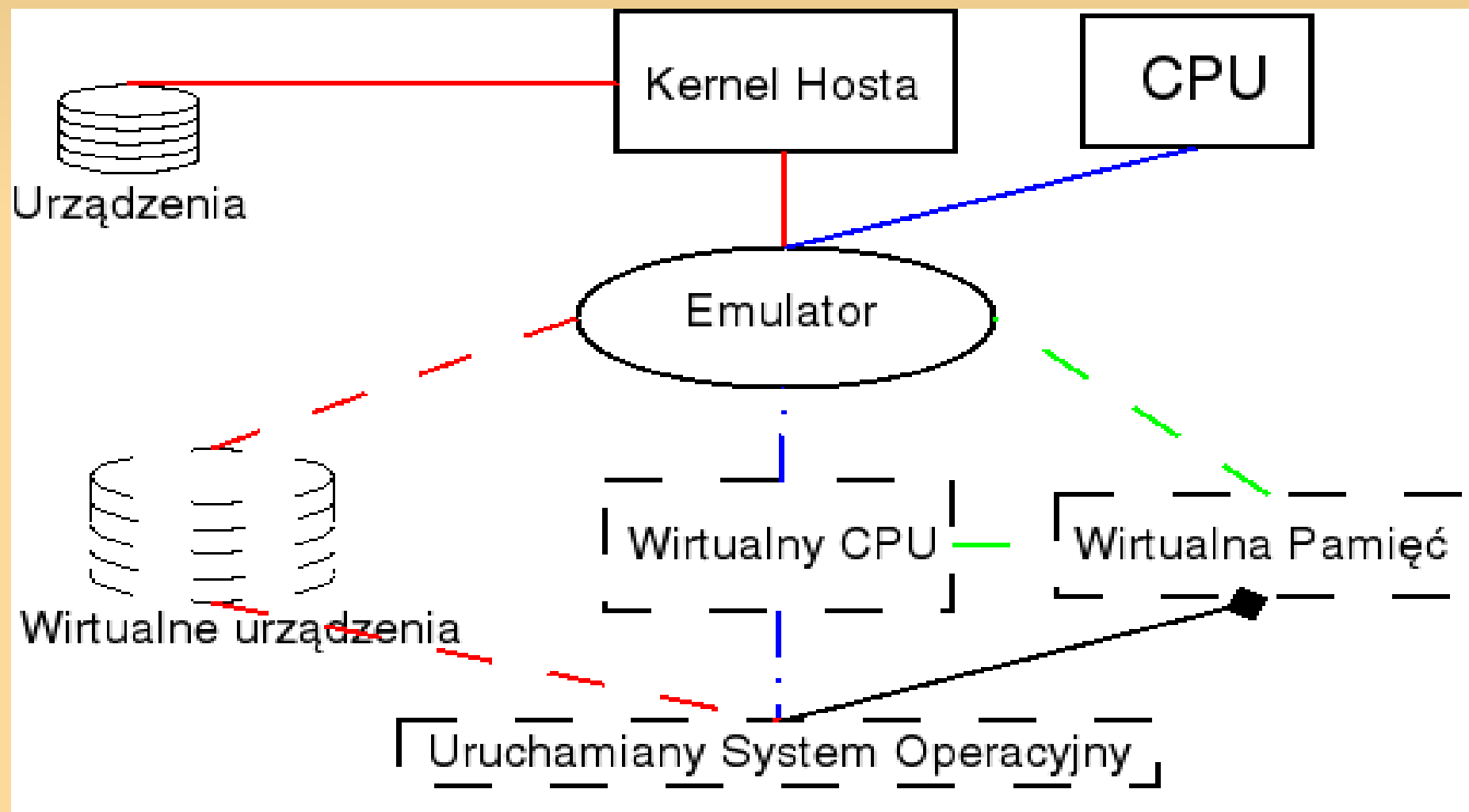
- emulację pełną (ew. czystą) – przykłady: Bochs, **QEMU**, emulatory ATARI, Amiga
- emulację API – przykłady: **Wine**, DOSBox, DOSEMU
- pełną (ew. właściwą) wirtualizację – przykłady: **VirtualPC**, **VMware**, Xen

Emulacja pełna..

polega na symulowaniu całego sprzętu na którym uruchamia się system który chcemy emulować. W szczególności program symuluje:

- pamięć
- procesor
- urządzenia I/O
- zegar
- i ogólnie wszystko

Emulacja pełna c.d.



Zalety pełnej emulacji

- Przenośność
- Pełna symulacja sprzętu
- Duża możliwość manipulacji sprzętem MW
- Nieograniczony dostęp do stanu maszyny
- Determinizm

Główne wady pełnej emulacji

- Przede wszystkim wydajność
- Duże zapotrzebowanie na zasoby

Warto zastopować maszynę kiedy się jej nie używa

- Utrudniona komunikacja między hostem a MW

Wirtualizacja właściwa

Idea:

- „Prawdziwe” uruchomienie wielu systemów na jednym komputerze. Prawdziwe w tym sensie, żeby operacje w ramach każdego systemu były faktycznie wykonywane bezpośrednio na procesorze.

Tylko czy sprzęt na to pozwala?

- Niestety na większości architektur nie

Problemy:

- Urządzenia zewnętrzne zakładające, że mają do czynienia z jednym sterownikiem.
- Sam procesor (IA32) jest zaprojektowany pod kątem jednego SO
 - segmentacja i stronicowanie
 - system ochrony pamięci
 - przerwania sprzętowe

Stosowane rozwiązanie – wirtualizacja sprzętu

- Bezpośrednio wykonujemy na procesorze 'bezpieczne' instrukcje
- Niebezpieczne musimy tłumaczyć
- Stosujemy „zwykłe” techniki emulacji sprzętu

Zalety

- Wirtualizacja udaje działanie całego komputera – guest nie ma jak zauważyć, że nie pracuje natywnie
- B. duża szybkość – kod guesta jest wykonywany prawie całkowicie bezpośrednio na procesorze
- Dzięki zrzuceniu na hosta odpowiedzialności za obsługę sprzętu można wirtualizować wiele urządzeń

Główne wady

- Trudna, zależna od architektury implementacja
- System gościa musi obsługiwać tę samą architekturę co system hosta (ale ograniczenie do x68 można jakoś przeżyć)