

Systemy operacyjne oparte na mikrojądrze

Grzegorz Anielak
Andrzej Kacperski
Remigiusz Mytyk

Uniwersytet Warszawski
12 grudnia 2008 r.

Spis treści

- Geneza i koncepcja
- Podział odpowiedzialności
- Komunikacja
- Wydajność
- Translatory
- Podsumowanie

Dlaczego mikrojądro?

- Wzrost liczby urządzeń dostępnych na rynku
- BSD – rok 1977
- Rozrost kodu

Koncepcja

- Część logiki systemu zaszyta w przestrzeni użytkownika
- Tylko najbardziej niezbędne elementy
- **MINIX, GNU/Hurd,** Amoeba, QNX, BeOS, *Singularity...*

MINIX

- Tworzony przez Andrew Tanenbauma w celach edukacyjnych



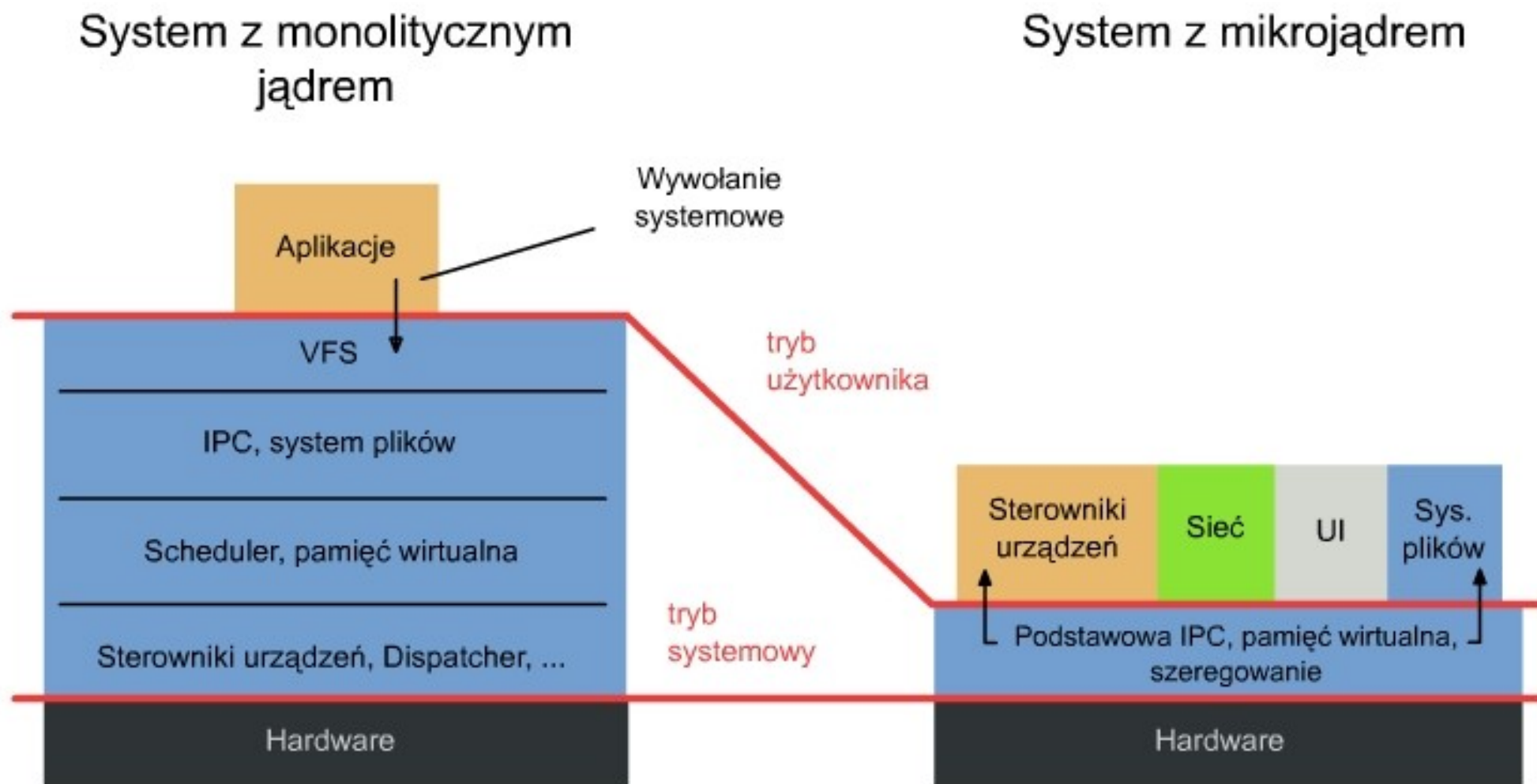
MINIX

- System uniksopodobny (*nix)
- 1987 – MINIX 1.0
- 2005 – MINIX 3.0

GNU/Hurd

- Rozwijany od 1984 roku
- Mikrojądro Mach 4
- Mechanizm ***translatorów***

Mikrojądro a jądro monolityczne



Samoleczenie

- Serwer reinkarnacji
 - Rodzic wszystkich procesów
 - Wznawia sterowniki i serwery
- Data store
 - Minibaza danych
 - Odzyskiwanie stanów po restarcie

Sposoby komunikacji

Wprowadzenie

W systemie Minix serwery oraz sterowniki działają jako niezależne procesy znajdujące się w przestrzeni użytkownika. Takie podejście sprawia, iż musimy sobie poradzić z dwoma zasadniczymi problemami:

- Zależności między modułami
- Ograniczone pole manewru w przestrzeni użytkownika

Sposoby komunikacji Zależności

- Sterowniki potrzebują informacji znajdujących się w przestrzeni jądra
- Sterowniki wywołują inne sterowniki
- Zadania wykonywane w trybie jądra zależne są od sterowników
- Sterowniki korzystają z I/O urządzeń zewnętrznych

Sposoby komunikacji IPC

- Komunikacja synchroniczna
- Wiadomości jednego typu i rozmiaru
- Kopiowane bezpośrednio pomiędzy procesami (brak buforowania)

Sposoby komunikacji

Blokady

- Gdy proces chce nadać wiadomość, a nie ma odbiorcy, bądź proces chce odebrać wiadomość, a nie ma nadawcy wtedy ten proces zostaje zablokowany..
- Czasem taka sytuacja jest nie do zaakceptowania, zwłaszcza gdy tym procesem jest jądro..

Sposoby komunikacji

Improved IPC

- Bitmapa zawierająca informacje z kim możemy się kontaktować
- Możliwość wymuszenia randki
- Nie blokujące, asynchroniczne IPC

Sposoby komunikacji Przerwania

- Przeniesienie logiki, która była związana z przerwaniami do modułów sterowników
- W przerwaniach jedynie niezbędne minimum
- Odbiór przerwania -> nadanie wiadomości

Sposoby komunikacji

Mechanizm powiadamiania

- Komunikacja asynchroniczna
- Brak blokowania
- Bitmapa powiadomień
- Gdy proces będzie gotowy jądro sporządzi wiadomość na podstawie powiadomienia i nadaje ją do odpowiedniego procesu

Sposoby komunikacji

Tryb użytkownika

- MMU czuwa by procesy zaglądały tylko do swojej przestrzeni adresowej
- Sterownikom to nie wystarcza
- Muszą prosić o pozwolenie

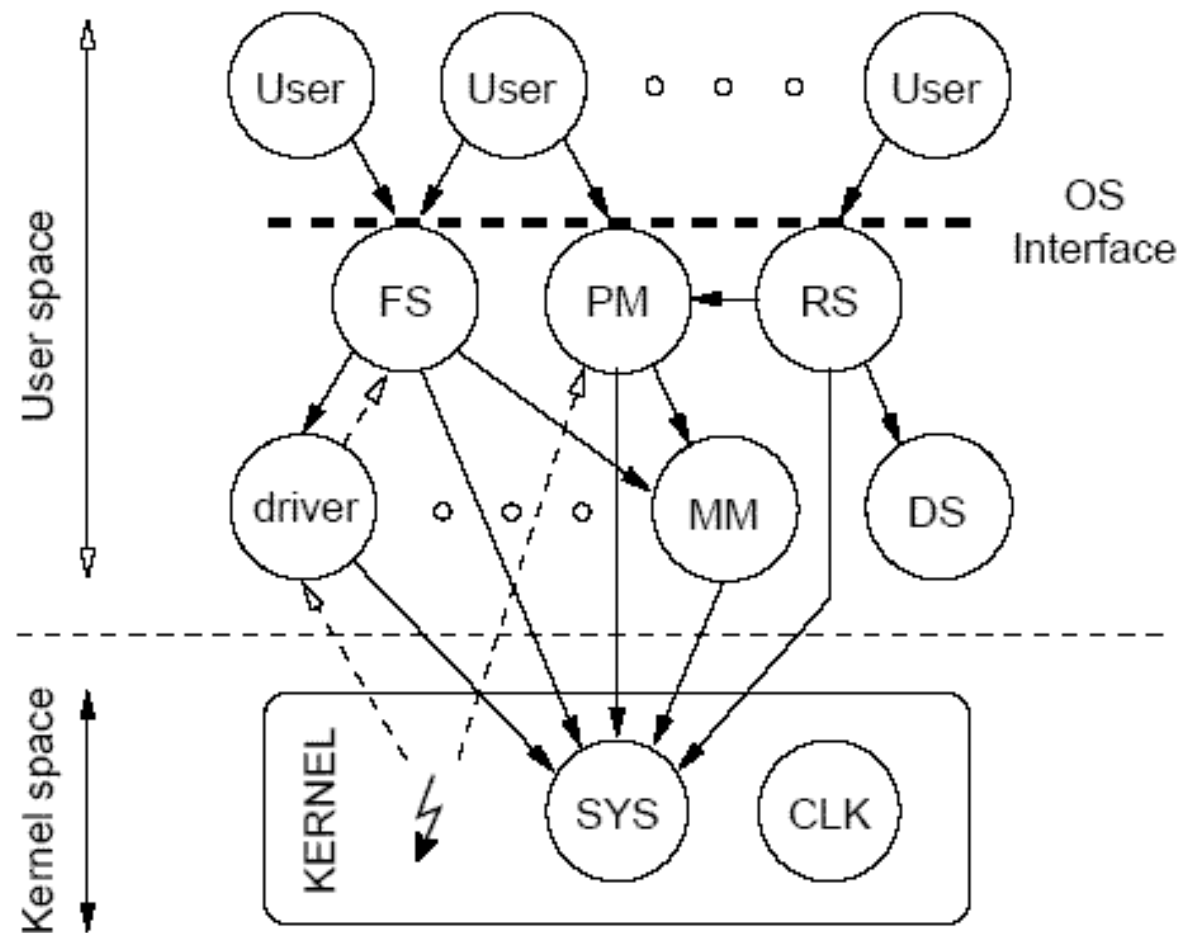
Sposoby komunikacji

Wywołania systemowe

- Wsparcie w postaci dodatkowego zestawu wywołań
- System task wydaje pozwolenia
- Jądro odpowiada wysyłając standardową wiadomość

Spôsoby komunikacj

Schemat



Sposoby komunikacji

Fork ()

- Funkcja biblioteczna fork() ->
- Wiadomość do Process Manager ->
- Ten pisze do Memory Manager
- A następnie do jądra
- Proces Manager odpowiada ->
- I następuje powrót z funkcji bibliotecznej

Wydajność

- Nie jest tak ważna
- Użytkownicy chętnie oddadzą 5%-10% wydajności w imię niezawodności
- ... tylko, że póki co nie jest to 5%-10%

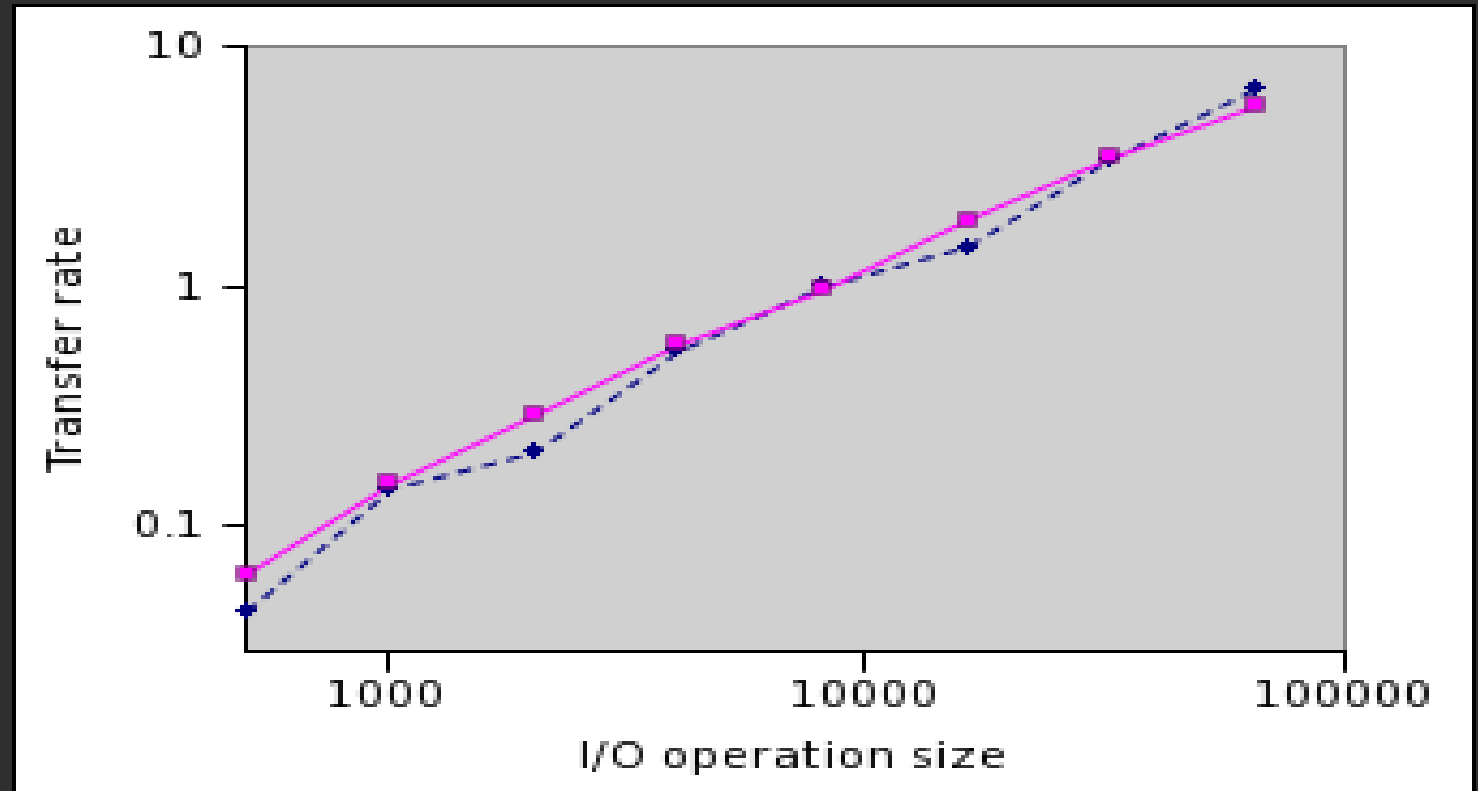
Dlaczego nie jest łatwo?

- Komunikaty: kopiowanie informacji zamiast ich współdzielenie
- Kosztowne zmiany kontekstu
- Trudniej wykorzystywać sprzęt

Benchmark

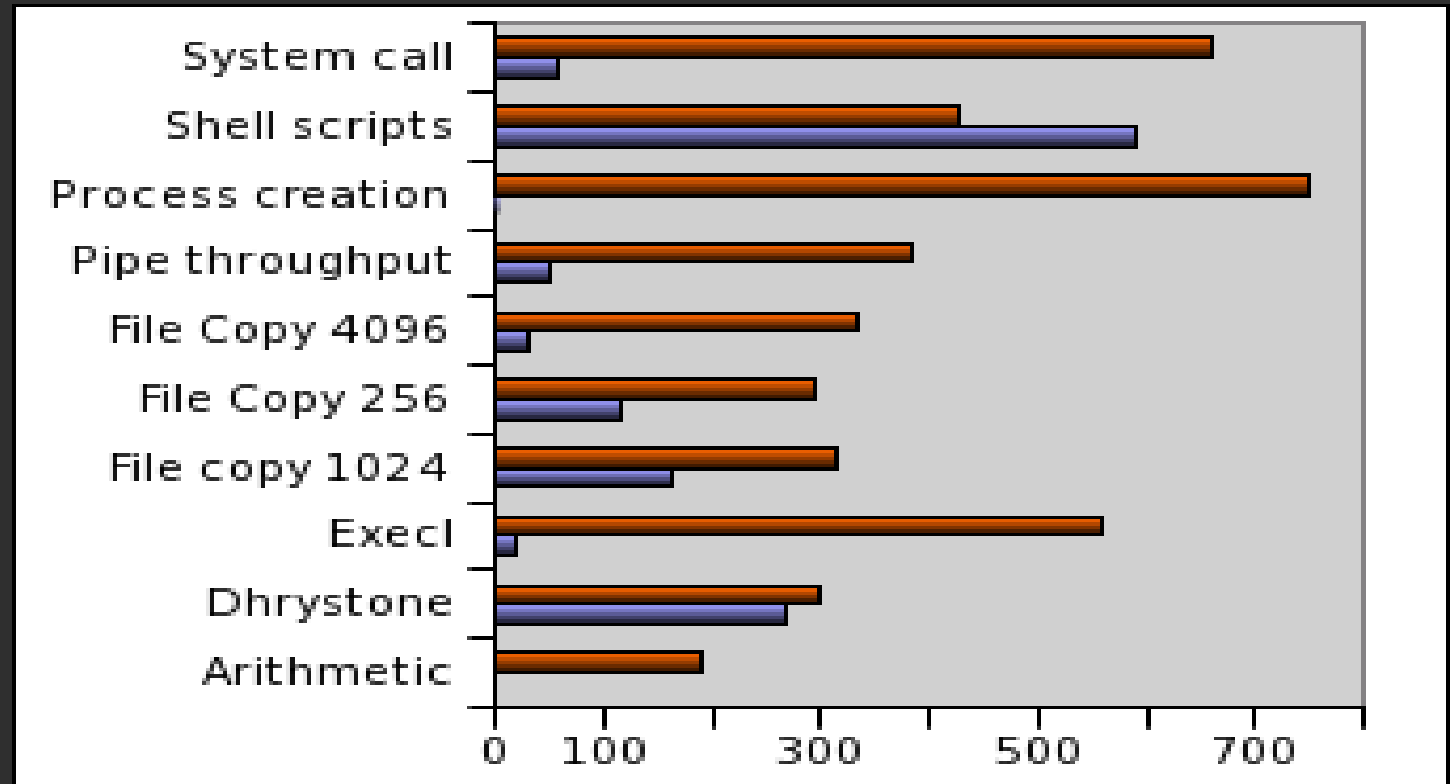
- lwn.net
- Minix 3 vs. Linux 2.6.17
- IOZone i UnixBench

IOZone



- Minix – linia przerywana
- Porównywalnie, ale..

UnixBench



- Minix 48 (niebieski)
- Linux 390 (pomarańczowy)
- więcej – lepiej, Minix fails

W obronie Minixa

- Tworzony przez grupkę ludzi
- Nie tak dojrzały jak Linux
- Mało stunningowany

Wnioski

- Linux jest fajny
- Minix na razie słaby
- Ostrożnie z mieszaniem dyskusji o ideach z dyskusją o implementacjach

Słowo o translatorach

- Główna idea Hurd-a
- Drzewo / - coś więcej niż drzewo katalogów
- Do punktu montowania doczepiamy program (translator) obsługujący żądania użytkownika

Translatory

- Pliki/katalogi abstrakcją wielu rzeczy: sterowniki urządzeń, zasobów sieciowych, dowiązań do plików, wielu innych...
- W Linux-ie FUSE

A vertical strip on the left side of the slide showing a blue sky with white clouds.

Demo

Dyskusja LT z AT (1992)

Argumenty za mikrojądrem:

- Niewiele wolniejsze, a niezawodne
- J. monolityczne to powrót do lat 70.
- Linux jest ściśle przywiązany do architektury 80x86
- Idealny dla studentów i badaczy

Dyskusja LT z AT (1992)

Argumenty przeciwko mikrojądru:

- **Teoria nie do końca poparta praktyką**
- **Nie ma wsparcia na żadnym sprzęcie**
- **Spore narzuty na komunikację procesów**
- **Trudności w programowaniu**

DZIĘKUJEMY
ZA UWAGĘ