

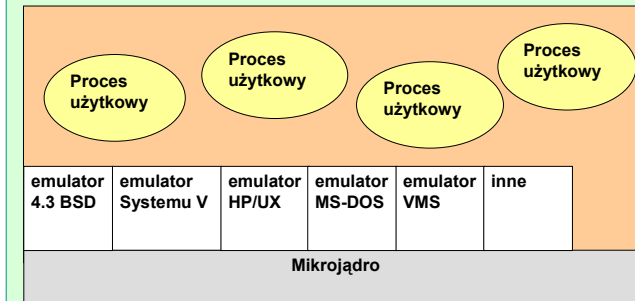
System operacyjny MACH

Zbigniew Suski

System operacyjny MACH

1

Emulacja w systemie MACH



Zbigniew Suski

System operacyjny MACH

2

Podstawowe cele projektu MACH

- ☐ Dostarczenie podstawy do budowy innych systemów operacyjnych.
- ☐ Udostępnianie wielkiej rozrzedzonej przestrzeni adresowej.
- ☐ Umożliwienie przezroczystego dostępu do zasobów sieciowych.
- ☐ Wykorzystanie równoległości zarówno w systemie jak i zastosowaniach.
- ☐ Uzyskanie przenośności w możliwie dużym zbiorze maszyn.

Zbigniew Suski

System operacyjny MACH

3

Jądro systemu

Podstawowe funkcje realizowane przez jądro:

- ☐ zarządzanie procesami,
- ☐ zarządzanie pamięcią,
- ☐ komunikacja,
- ☐ zarządzanie usługami wejścia/wyjścia.

Podstawowe abstrakcje urzeczywistniane przez jądro:

- ☐ procesy lub zadania (*tasks*),
- ☐ wątki (*threads*),
- ☐ porty (*ports*),
- ☐ komunikaty (*messages*),
- ☐ obiekty pamięci (*memory objects*).

Zbigniew Suski

System operacyjny MACH

4

Elementy procesu

- ❑ Przestrzeń adresowa
- ❑ Wątki wykonywane w przestrzeni adresowej
- ❑ Porty

- Proces wywołuje operacje jądra poprzez wysłanie do niego odpowiedniego komunikatu.
- Operacja tworzenia procesu określa proces prototypowy, którym nie musi być proces tworzący.

Zbigniew Suski

System operacyjny MACH

5

Wątki

- ❑ Wykonują instrukcje i działają na swoich rejestrach oraz przestrzeni adresowej.
- ❑ Współdzielą przestrzeń adresową procesu i wszystkie zasoby ogólnoprocesowe.
- ❑ Mają również swoje własne, prywatne zasoby.
- ❑ Porty są znane w całym procesie.
- ❑ Tworzenie i usuwanie wątków jest wykonywane przez jądro.
- ❑ Wątek podobnie jak proces może być zdalny do działania lub zablokowany.

Zbigniew Suski

System operacyjny MACH

6

Operacje na wątkach

- ❑ Tworzenie wątku
- ❑ Zakończenie działania wątku
- ❑ Blokowanie wątku w celu oczekiwania na zakończenie działania przez wątek potomny
- ❑ Oświadczenie, że dany wątek nie będzie oczekiwany przez inny
- ❑ Przekazanie informacji dla planisty, że wątek oczekuje na zdarzenie i może zostać podjęty inny wątek
- ❑ Pobranie identyfikatora
- ❑ Funkcje synchronizacji za pomocą zamków i zmiennych warunkowych
- ❑ Działania na zmiennych warunkowych (blokowanie i budzenie wątków)

Zbigniew Suski

System operacyjny MACH

7

Planowanie

Zbiór procesorów (*processor set*)

- jednostki przetwarzające
- wątki

Priorytety wątku:

- ❑ **podstawowy** - ustawiany w pewnych granicach przez sam wątek
- ❑ **graniczny** - najmniejsza dopuszczalna wartość (najwyższy priorytet) priorytetu podstawowego
- ❑ **bieżący** - wykorzystywany w planowaniu; obliczany jako suma priorytetu podstawowego i wartości będącej funkcją ostatniego zatrudnienia jednostki centralnej przez wątek

Zbigniew Suski

System operacyjny MACH

8

Algorytm planowania

- ❑ Globalna tablica kolejek wykonań
- ❑ Lokalne tablice kolejek wykonań
- ❑ Zmienność kwantu czasu
- ❑ Zmienność priorytetu
- ❑ Planowanie swobodne (*handoff scheduling*)
- ❑ Pokrewieństwo

Zbigniew Suski

System operacyjny MACH

9

Zarządzanie pamięcią

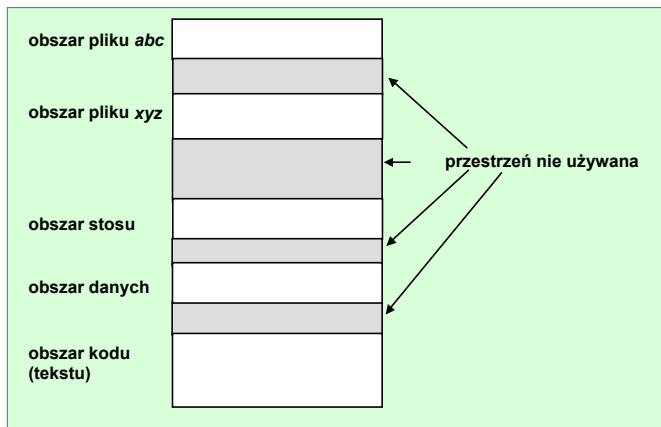
- ❑ Moduł pmap wykonywany w jądrze i odpowiedzialny za działanie MMU.
- ❑ Maszynowo niezależny kod jądra. Przetwarza zdarzenia braku stron, zarządza mapami adresów i zastępowania stron.
- ❑ Proces użytkowy zwany zarządcą pamięci (*memory manager*) lub zewnętrzną procedurą stronicującą (*external pager*).

Zbigniew Suski

System operacyjny MACH

10

Struktura przestrzeni adresowej



Zbigniew Suski

System operacyjny MACH

11

Zarządzanie pamięcią

- ❑ Obszary (*regions*)
- ❑ Obiekty pamięci (*memory objects*)
- ❑ Atrybut dziedziczności (*inheritance attribute*)
 - Obszar nie jest używany w procesie potomnym
 - Obszar jest dzielony przez proces prototypowy i potomny
 - Obszar w procesie potomnym jest kopią prototypowego

Zbigniew Suski

System operacyjny MACH

12

Zarządzanie pamięcią

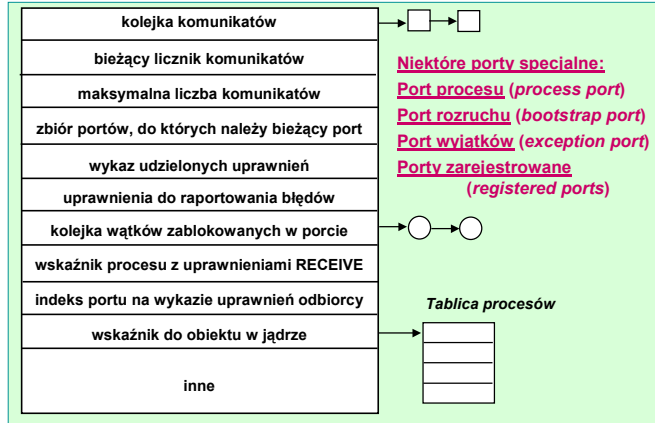
- ❑ Zarządcy pamięci
 - Port obiektu (*object port*)
 - Port nadzorczy (*control port*)
 - Port nazwy (*name port*)
- ❑ Demon stronicowania
- ❑ Strona dzielona i serwer DSM (*Distributed Shared Memory Server*)

Zbigniew Suski

System operacyjny MACH

13

Komunikacja – struktura portu

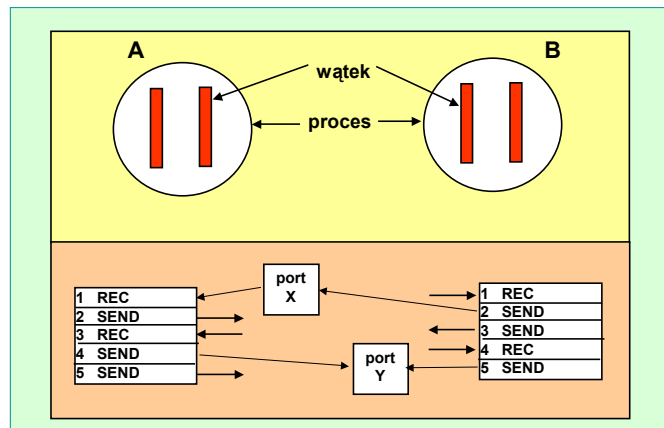


Zbigniew Suski

System operacyjny MACH

14

Uprawnienia dostępu do portów

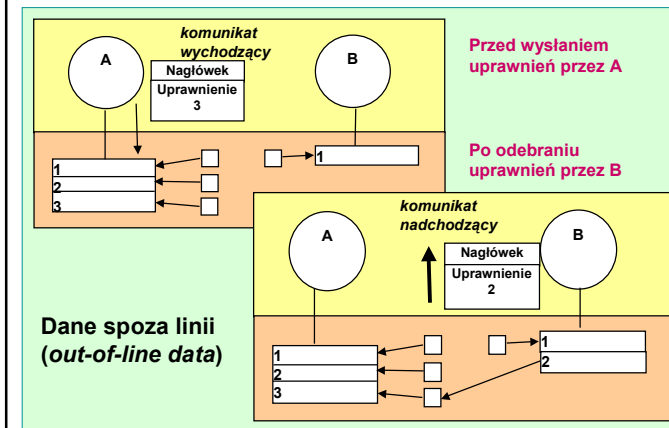


Zbigniew Suski

System operacyjny MACH

15

Zmiana uprawnień dostępu do portów w locie

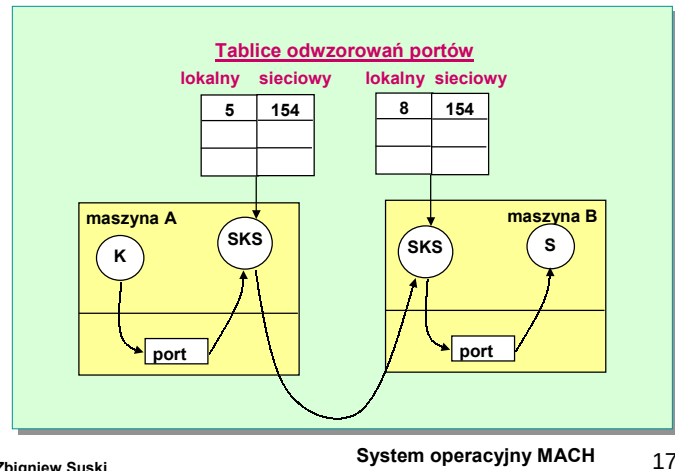


Zbigniew Suski

System operacyjny MACH

16

Komunikacja międzymaszynowa



Zbigniew Suski

System operacyjny MACH

17

Emulacja systemu UNIX

Implementacja emulacji systemu UNIX składa się z dwóch części:

- serwera systemu UNIX
- biblioteki emulującej odwołania do systemu

- ❑ Jądro MACHa wychwytyje wszelkie odwołania do systemu UNIX i przenosi je do procedur bibliotecznych emulujących wywołania systemowe.
- ❑ Powrót do przestrzeni użytkownika przy użyciu mechanizmu trampoliny (*trampoline mechanism*).

Zbigniew Suski

System operacyjny MACH

18

Emulacja systemu UNIX

- ❑ Potomkowie procesu *init*, dziedziczą bibliotekę emulującą i mechanizm trampoliny.
- ❑ Wykonanie funkcji EXEC powoduje zastąpienie w przestrzeni adresowej tylko części stanowiącej program uniksowy. Nie jest zastępowana biblioteka emulująca.
- ❑ Zgłoszenia do serwera systemu UNIX przekazywane są w formie komunikatów.
- ❑ Otwierany plik zostaje odwzorowany bezpośrednio w przestrzeni adresowej procesu wywołującego. Biblioteka emulująca ma do niego dostęp bezpośredni i nie musi wykonywać zdalnych odwołań do serwera uniksowego.

Zbigniew Suski

System operacyjny MACH

19