

## Systemy rozproszone

**System rozproszony** jest zbiorem niezależnych komputerów (aspekt sprzętowy), które z punktu widzenia użytkowników systemu sprawiają wrażenie pojedynczego komputera (aspekt programowy)

**System rozproszony** jest zbiorem komputerów, które nie mają wspólnej pamięci ani zegara

**System rozproszony** jest zbiorem niezależnych komputerów połączonych siecią komunikacyjną, która umożliwia wymianę komunikatów

**System rozproszony** to termin służący do określenia szerokiej gamy systemów komputerowych od *systemów słabo powiązanych* takich jak *sieci rozległe*, poprzez systemy *silnie powiązane* takie jak *sieci lokalne*, do systemów *bardzo silnie powiązanych* takich jak *systemy wieloprocesorowe*

### Zalety systemów rozproszonych

- *Dzielenie zasobów* (dane, urządzenia sprzętowe jak np. drukarki, dyski)
- *Przyśpieszenie obliczeń* (dzielenie obciążenia)
- *Niezawodność* (awaria jednego urządzenia nie powinna uniemożliwiać działania systemu, lecz co najwyżej pogorszyć jego wydajność)
- *Komunikacja* (np. poczta elektroniczna)
- *Ekonomiczność* (system rozproszony może być tańszy niż odpowiadający mu mocą obliczeniową system scentralizowany)

- *Wewnętrzne rozproszenie* (niektóre aplikacje są z natury rozproszone i wymagają rozproszonych komputerów)
- *Stopniowy wzrost* (można stopniowo zwiększać moc obliczeniową systemu; *skalowalność* to zdolność systemu do adaptowania się do wzrastających zapotrzebowań)

### Wady systemów rozproszonych

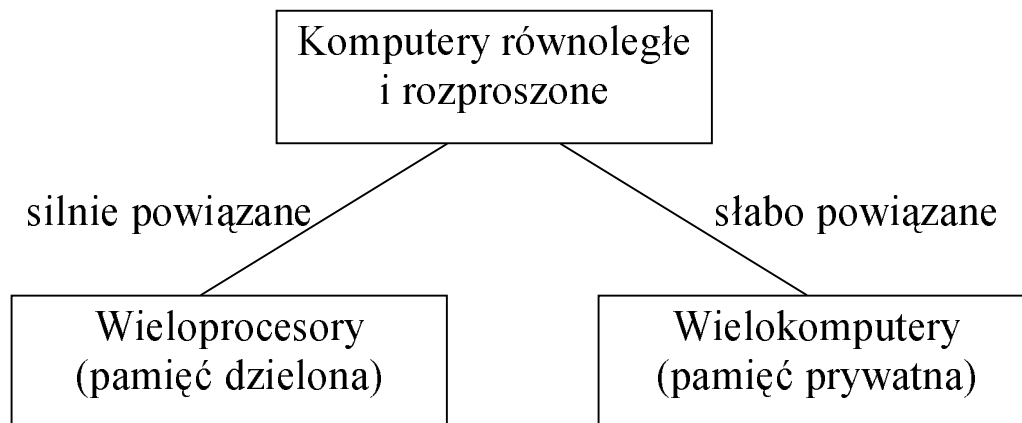
- *Oprogramowanie* (zdecydowanie bardziej złożone; wymaga opracowania wspólnych standardów)
- *Sieć* (może ulec awarii lub zostać przeciążona)
- *Bezpieczeństwo* (komputer podłączony do sieci jest mniej bezpieczny)

### Systemy rozproszone a systemy równoległe

*Klasyfikacja systemów wielokomputerowych wg Flynna:*

- SISD** Single Instruction, Single Data Stream  
**SIMD** Single Instruction, Multiple Data Streams  
(systemy wektorowe)  
**MISD** nie ma  
**MIMD** Multiple Instruction, Multiple Data Stream

*Klasyfikacja architektur MIMD:*



## Cechy systemów rozproszonych

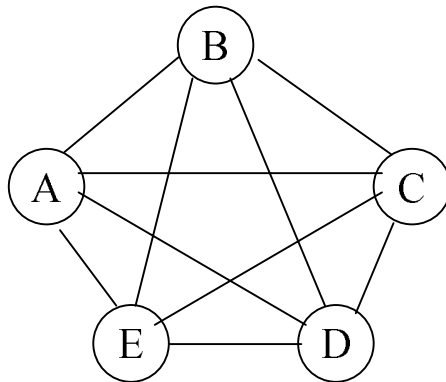
- **Przezroczystość** (ang. *transparency*)
  - *położenia* (zasobów sprzętowych i programowych)
  - *migracji* (zasoby, np. pliki, mogą migrować, czyli zmieniać położenie bez zmiany nazwy)
  - *istnienia kopii* (zasoby mogą istnieć w wielu kopiach dla zwiększenia wydajności i/lub niezawodności)
  - *błędu* (program zostanie dokończony bez względu na awarie)
  - *współużytkowania* (użytkownicy mogą wspólnie korzystać z zasobów)
  - *równoległości* (obliczenia mogą przebiegać równoległe)
- **Niejednorodność** (ang. *heterogeneity*)
  - *sprzętu komputerowego* (różne architektury węzłów systemu rozproszonego)
  - *sieci* (różne środki transmisji, sygnały, interfejsy sieciowe, protokoły)
  - *oprogramowania*
- **Autonomiczność** (ang. *autonomy*)
  - *wysoka* (redundancja, niskie współużytkowanie zasobów, może wymagać centralnego sterowania)
  - *niska* (mniejsza elastyczność, większa zawodność)

## Topologia systemu rozproszonego

Kryteria oceny:

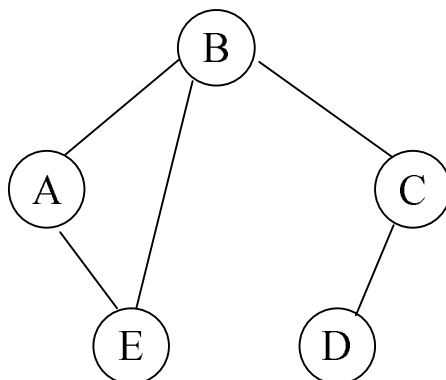
- *koszt podstawowy* (ile kosztuje połączenie różnych stanowisk w jeden system?)
- *koszt komunikacji* (ile trwa przesłanie komunikatu z jednego stanowiska do innego?)
- *niezawodność* (jeśli łącze lub stanowisko ulegnie awarii, to czy pozostałe stanowiska pozostaną w kontakcie?)

## Połączenie pełne



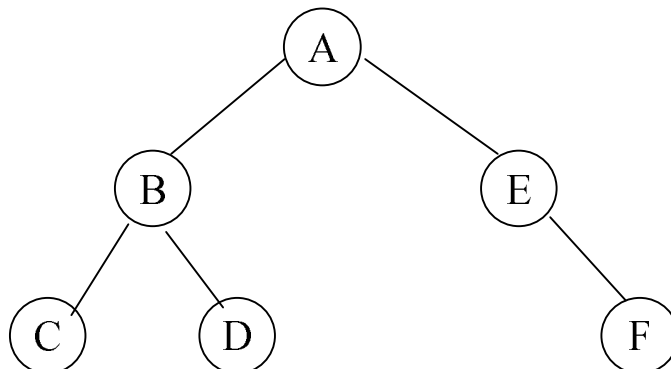
- *koszt podstawowy*: rośnie wprost proporcjonalnie do kwadratu liczby stanowisk
- *koszt komunikacji*: komunikat pokonuje tylko jedno łącze
- *niezawodność*: b. duża

## Połączenie częściowe



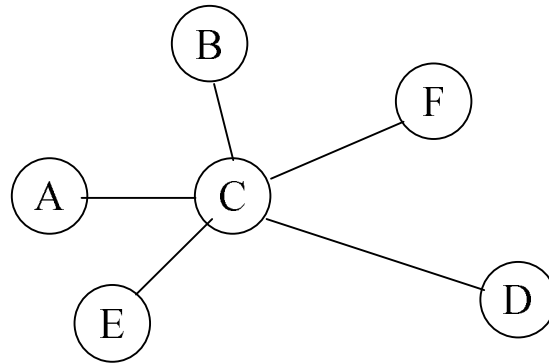
- *koszt podstawowy*: niższy niż dla sieci pełnej
- *koszt komunikacji*: wysłanie komunikatu może wymagać przejścia przez kilka stanowisk pośrednich
- *niezawodność*: awaria jednego łącza może rozbić sieć

## Hierarchia



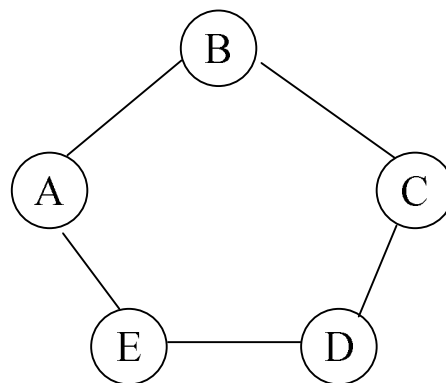
- *koszt podstawowy*: na ogół mniejszy niż dla sieci częściowej
- *koszt komunikacji*: wyższy niż dla sieci pełnej
- *niezawodność*: awaria dowolnego węzła (z wyjątkiem liścia) powoduje na ogół rozbitcie sieci

### Gwiazda



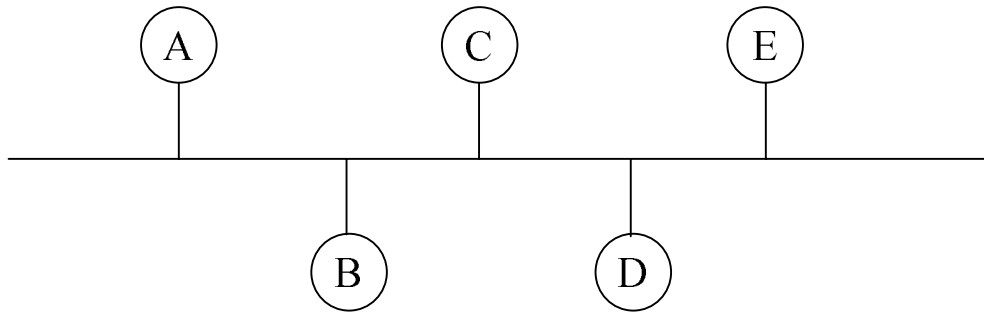
- *koszt podstawowy*: funkcja liniowa liczby stanowisk
- *koszt komunikacji*: niski, ale węzeł centralny może być wąskim gardłem
- *niezawodność*: awaria stanowiska centralnego powoduje rozbitcie sieci

### Pierścień



- *koszt podstawowy*: funkcja liniowa liczby stanowisk
- *koszt komunikacji*: może być wysoki
- *niezawodność*: awaria stanowiska (lub łącza) powoduje rozbitcie sieci

## Szyna wielodostępna



- *koszt podstawowy*: funkcja liniowa liczby stanowisk
- *koszt komunikacji*: dość niski
- *niezawodność*: awaria stanowiska nie zakłóca komunikacji, awaria szyny powoduje rozbitcie sieci

## Typy sieci

- Sieci lokalne (ang. *LAN*) - obejmują niewielki obszar geograficzny
  - szyna wielodostępna, sieć pierścieniowa lub gwiazdzista
  - prędkość rzędu 10 Mbitów/s lub wyższa
  - opóźnienia komunik. i współcz. błędów są niewielkie
  - wspólne korzystanie z zasobów b. naturalne
  - rozgłaszanie jest tanie i szybkie
- Sieci globalne (ang. *WAN*) - obejmują duży obszar geograficzny
  - linie telefoniczne, łącza mikrofalowe, kanały satelitarne
  - prędkość rzędu 100 Kbitów/s
  - duże opóźnienia i częste błędy w transmisji
  - złożone systemy zarządzania
  - rozgłaszanie zwykle wymaga wielu komunikatów

## Komunikacja w systemach rozproszonych

1. *Strategie wyboru trasy*: które komunikaty są przesyłane przez sieć?

- trasa stała (ang. *fixed routing*) - droga od A do B jest ustalona i zmienia się jedynie w przypadku sprzętowej awarii; zwykle najkrótsza, brak dostosowania do zmian obciążenia, komunikaty dotrą w kolejności wysłania
- trasa wirtualna (ang. *virtual circuit*) - drogę z A do B ustala się na okres jednej sesji; możliwość dostosowania do zmian obciążenia, komunikaty dotrą w kolejności wysłania
- trasa dynamiczna (ang. *dynamic routing*) - drogę z A do B wybiera się w chwili wysyłania komunikatu; zwykle wybiera się najmniej obciążone łącze, komunikaty mogą docierać w innej kolejności - można je numerować

2. *Strategie połączeń*: jak dwa procesy wysyłają ciąg komunikatów?

- komutowanie łączy (ang. *circuit switching*) - stałe fizyczne połączenie na czas trwania komunikacji; analogia → połączenie telefoniczne
- komutowanie komunikatów (ang. *message switching*) - ustala się czasowe łącze na czas przesyłania jednego komunikatu
- komutowanie pakietów (ang. *packet switching*) - komunikaty zmiennej długości dzieli się na pakiety o stałym rozmiarze, każdy pakiet może być wysłany inną drogą, pakiety trzeba ponownie łączyć w komunikaty w miejscu przeznaczenia

### 3. *Współzawodnictwo*: jak rozwiązywać konflikty wynikające z współużytkowania sieci?

- CSMA/CD (ang. *Carrier sense with multiple access and collision detection*) - stanowisko prowadzi nasłuch łącza; jeśli łącze wolne to rozpoczyna transmisję (dalej nasłuchując); jeśli wykryje kolizję, to przestaje nadawać i generuje sygnał informujący o kolizji, po losowym czasie próbuje znowu; niebezpieczeństwo wielu kolizji w przeciążonym systemie; standard IEEE 802.3; używane w Ethernetie
- przekazywanie żetonu (ang. *token passing*) - komunikat-żeton krąży stale w sieci (zwykle pierścień); stanowisko, które chce wysłać komunikat czeka na żeton, usuwa go z sieci, transmituje komunikat i ponownie przesyła żeton; standard IEEE 802.5
- przegródki na komunikaty (ang. *message slots*) - pewna liczba stałej długości przegródek na komunikaty stale krąży w sieci (zwykle pierścień); stanowisko czeka aż nadejdzie pusta przegródka, wstawia do niej komunikat (ew. podzielony na pakiety) i informację kontrolną

### 4. *Nadawanie nazw i rozpoznawanie nazw*: jak dwa procesy mogą „odszukać” się w sieci?

**DNS (Domain Name Service)** - określa sposób nadawania nazw komputerom macierzystym w sieci Internet

## Protokoły komunikacyjne

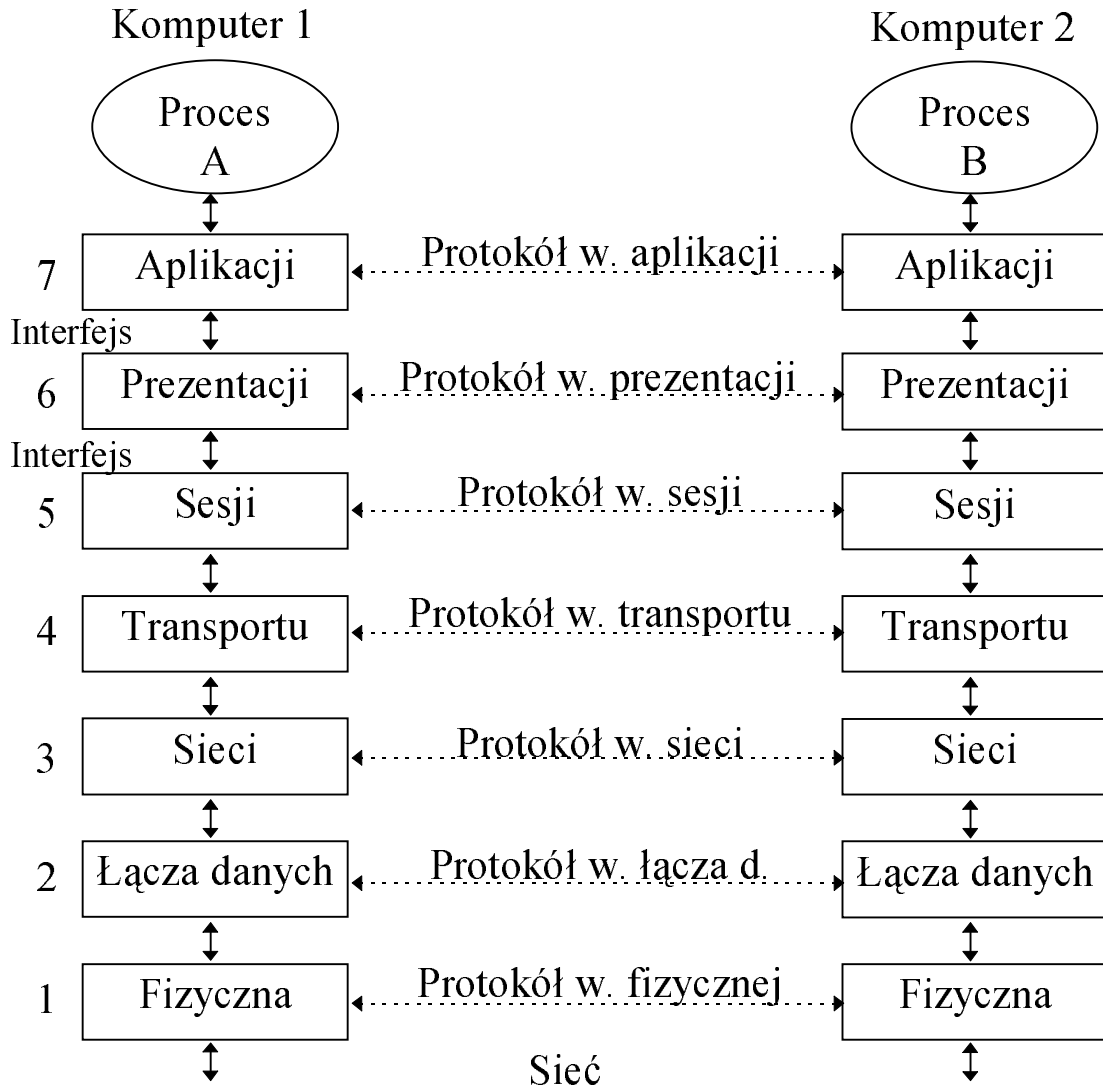
**Protokół** - zbiór formalnych reguł i konwencji dotyczących formatu i synchronizacji w czasie wymiany komunikatów między procesami

**Architektura sieci**: podział na warstwy i protokoły



# *Open System Interconnection (OSI) Reference Model of the International Organization for Standardization (ISO)*

- czyli model ISO/OSI



## 1. Warstwa fizyczna

Odpowiada za fizyczną transmisję bitów między dwoma warstwami łącza danych. Zajmuje się synchronizacją na poziomie bitów i sygnałami zależnymi od środka transmisji (jak fizycznie wygląda bit 1 i bit 0, jaka jest prędkość transmisji w bitach/s, czy transmisja może przebiegać równocześnie w dwóch kierunkach itp.)

## 2. Warstwa łączy danych

Zajmuje się grupowaniem bitów w ramki i ich transmisją, wykrywaniem i usuwaniem błędów (tworzy sumy kontr.)

## 3. Warstwa sieci

Zajmuje się ustalaniem tras pakietów w sieci, organizacją połączeń, obsługą adresów, utrzymywaniem informacji o trasach i reagowaniem na zmieniające się obciążenie.

Protokół połączeniowy: **X.25**, protokół bezpołączeniowy: **IP**

## 4. Warstwa transportu

Dostarcza warstwie sesji niezawodne połączenie.

Odpowiada za przesyłanie komunikatów między klientami, dzielenie komunikatów na pakiety i łączeniem pakietów w komunikaty, pilnowaniem kolejności pakietów, generowaniem adresów fizycznych. Protokół połączeniowy: **TCP** i bezpołączeniowy: **UDP**

## 5. Warstwa sesji

Ulepszona wersja warstwy transportu. Odpowiada za implementację sesji. Na tym poziomie realizuje się RPC

## 6. Warstwa prezentacji

Odpowiada za pokonywanie różnic w formatach danych między różnymi stanowiskami w sieci, za kompresję i kodowanie/dekodowanie danych, za konwersję znaków

## 7. Warstwa aplikacji

Odpowiada za bezpośrednią interakcję z użytkownikami, przesyła pliki, obsługuje telnet, pocztę elektroniczną, rozproszone bazy danych

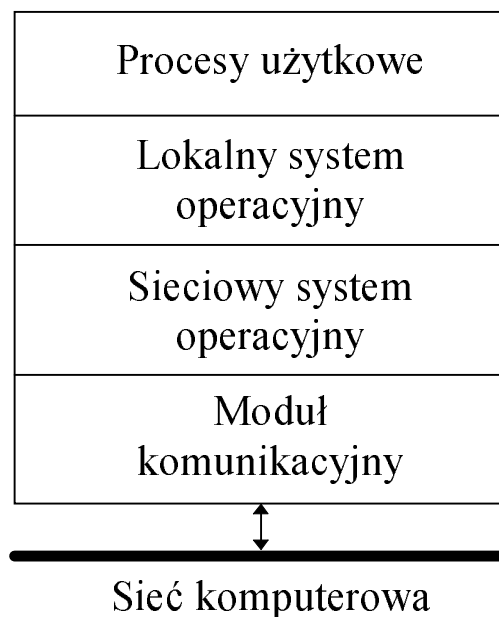
- Każdy komputer w sieci komunikacyjnej musi zapewniać trzy najniższe warstwy
- Każda warstwa przed wysłaniem dołącza do przesyłanej informacji pewien *nagłówek* (ew. *stopkę*), a po odebraniu - usuwa (**kapsułkowanie**, ang. *encapsulation*)

1: strumień bitów, 2: ramka, 3: pakiet, 4 i 5: komunikat, 6: zaszyfrowane dane, 7: dane użytkownika

- Model ISO/OSI jest b. ogólny. Każda warstwa to dodatkowy narzut na transmisję. Często implementuje się mniejszą liczbę warstw (np. TCP/IP - cztery warstwy)

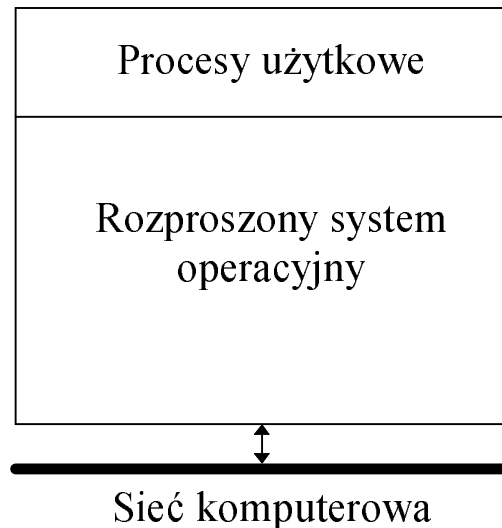
## Klasyfikacja systemów operacyjnych dla SR

### 1. Sieciowy system operacyjny



- każdy komputer ma swój prywatny system operacyjny
- użytkownik wie, na którym komputerze pracuje, użycie zdalnego komputera wymaga jawnego zarejestrowania się na nim (rlogin, telnet)
- użytkownik wie, gdzie znajdują się jego pliki, musi jawnie przesyłać pliki między komputerem zdalnym i lokalnym (FTP)
- mała tolerancja na błędy
- przykład: **NFS** (Network File System firmy Sun)

## 2. Rozproszony system operacyjny



- słabo powiązany sprzęt, silnie powiązane oprogramowanie
- jeden system operacyjny
- użytkownicy nie muszą być świadomi liczby komputerów ani położenia swoich plików
- dostęp do zasobów zdalnych uzyskują tak samo jak do lokalnych
- przemieszczanie danych, obliczeń i procesów z jednego komputera na drugi odbywa się pod nadzorem SO
  - przemieszczanie danych: można przemieszczać całe pliki lub ich fragmenty
  - przemieszczanie obliczeń: czasem bardziej opłaca się przesłać obliczenia zamiast danych
  - przemieszczanie procesów: równoważenie obciążenia, przyśpieszenie obliczeń, preferencje sprzętowe, preferencje oprogramowania; migracja procesu może być jawna lub niejawną
- komputery w systemie rozproszonym nie są autonomiczne
- duża tolerancja na błędy
- przykład: Amoeba, Mach, Chorus, DCE,

Różnicę między systemem sieciowym i rozproszonym można wyjaśnić na podstawie położenia modułu IPC w hierarchicznej strukturze systemu operacyjnego: im moduł znajduje się niżej, tym bardziej przezroczysta staje się sieć, decyzje można podejmować globalnie, a zasobami można zarządzać za pomocą globalnych mechanizmów

Zarządzanie plikami
IPC
Zarządzanie urządzeniami we-
Zarządzanie pamięcią
Zarządzanie procesami

Operacje na plikach

Zarządzanie plikami
Zarządzanie urządzeniami we-
IPC
Zarządzanie pamięcią
Zarządzanie procesami

Zdalne operacje na plikach

Zdalne drukowanie  
Zdalne korzystanie z plotera

Zarządzanie plikami
Zarządzanie urządzeniami we-
Zarządzanie pamięcią
Zarządzanie procesami i IPC

Zdalne operacje na plikach

Zdalne drukowanie  
Zdalne korzystanie z plotera

Zdalne tworzenie procesów,  
niszczenie, szeregowanie,  
synchronizacja, ochrona