Webserver

Programmering af tråde

Fil system

BSD

NAS

FreeNAS

Operativsystemer og netværk

Lektion 6

Webserver

Hvordan ser en webserver ud

Webserver

- En webserver er et program, der bruger HTTP (Hypertext Transfer Protocol) til at serve de filer, der danner websider til brugerne, som svar på deres anmodning, som er fremsendt af deres computeres HTTP-klienter.
- Processen er et eksempel på klient / server-modellen. Alle computere der er værter for websteder skal have Web server-programmer kørende.
- Førende webservere omfatter Apache (den mest udbredt installerede webserver), Microsofts Internet Information Server (IIS) og nginx (udtales engine X) fra NGNIX. Andre webservere omfatter Novells NetWare-server, Google Web Server (GWS) og IBMs familie af Domino-servere.

Webserver

- Webservere ofte kommer som en del af en større pakke af internet og intranet-relaterede programmer til serving af email, download anmodninger om File Transfer Protocol (FTP) filer, og værktøjer til at bygge- og udgivelse websider.
- Overvejelser om at vælge en webserver omfatter hvor godt den virker med operativsystemet og andre servere, dens evne til at håndtere server-side programmering, sikkerheds egenskaber og de særlige udgivelses, søgemaskine og site bygning værktøjer, der kommer med den.

Webserver

The files stored on web servers are read by browsers (such as Firefox, Safari, Chrome or Internet Explorer) which convert these files into images and text for you to view. Your browser communicates with web servers to bring you information from the internet.

Webserver Software

- Der er mange valg der skal foretages når man beslutter at sætte en web-server op fra bunden, da det definerer alle muligheder senere hen for både brugeren og administratoren af systemet.
- Inden man træffer de mest grundlæggende valg bør man sætte sig fast på om ens web-service skal fokusere på asp.net eller php, da disse kører optimalt på forskellige operativ-system platforme; Windows Server og afarter af Unix systemer respektivt. Der er løsninger, så de hver især kan køre "det andet" sprog, se f.eks. Mono, men de understøtter ikke løsningerne fuldt ud og en native platform er derfor at foretrække hvis man ikke har strengt behov for begge platforme på samme server.

Webserver Software

- Der findes en del forskellige webservere til Linux, men de to mest populære er Apache og NGINX.
- Apache er klart den mest udbredte med knap tre gange så mange aktive sider som nummer to, NGINX (49% og 17% respektivt).
- Apache projektet stammer helt tilbage fra 1995 og understøtter en lang række sprog som Perl, Python, Tcl og PHP.
- Man kan udvide kernefunktionaliteten i Apache med moduler.
- Den store svaghed ved Apache er at den kan være mere resurse krævende, da hver forbindelse har en separat proces, hvilket gør at den kræver mere hukommelse ved høj server belastning da hver proces indeholder overhead for PHP og andre moduler man har indlæst.

Webserver Software

- NGINX er meget nyere, skabt af Igor Sysoev i 2002, og siden den er nyere var den fra starten af lavet ud fra en opmærksomhed omkring de problemer der kan komme ved mange forbindelser.
- Således bruger den en asynkron, ikke-blokerende forbindelses behandlings algoritme der benytter Linux' "epoll" API.
- På den måde kan én proces behandle mange forbindelser, hvilket fungerer godt sammen med PHP-fortolkeren phpfpm.
- NGINX har modsat ikke Apache fortolkere indbygget, så den må ty til eksterne der så optager forbindelses-pladser.

Webserver Kontrolpaneler

- Hosting kontrolpaneler er web-baserede grænseflader der tillader administratorer og / eller brugere til at styre forskellige server-tjenester fra komforten i en webbrowser.
- Web hosting control panel software kan give adgang til:
  - Domain name system management (web domains, mail domains, etc.)
  - Email system management (email addresses, email kvoter, spam forebyggelse, etc.)
  - FTP management (user accounts, password management, file system quotas)
  - Web-baseret fil system adgang
  - SSH bruger/nøgle management
  - Database management (MySQL / MariaDB, PostgreQSL og nogle gange andre database systemer)
  - Backup management
  - Logfile adgang og rapportering
  - Plugin system til at konfigurere yderligere services og installing af apps (f.eks. WordPress)

Webserver Kontrolpaneler

- Hosting kontrolpaneler er web-baserede grænseflader der tillader administratorer og / eller brugere til at styre forskellige server-tjenester fra komforten i en webbrowser.
- Web hosting control panel software kan give adgang til:
  - Domain name system management (web domains, mail domains, etc.)
  - Email system management (email addresses, email kvoter, spam forebyggelse, etc.)
  - FTP management (user accounts, password management, file system quotas)
  - Web-baseret fil system adgang
  - SSH bruger/nøgle management
  - Database management (MySQL / MariaDB, PostgreQSL og nogle gange andre database systemer)
  - Backup management
  - Logfile adgang og rapportering
  - Plugin system til at konfigurere yderligere services og installing af apps (f.eks. WordPress)

Webserver Kontrolpaneler

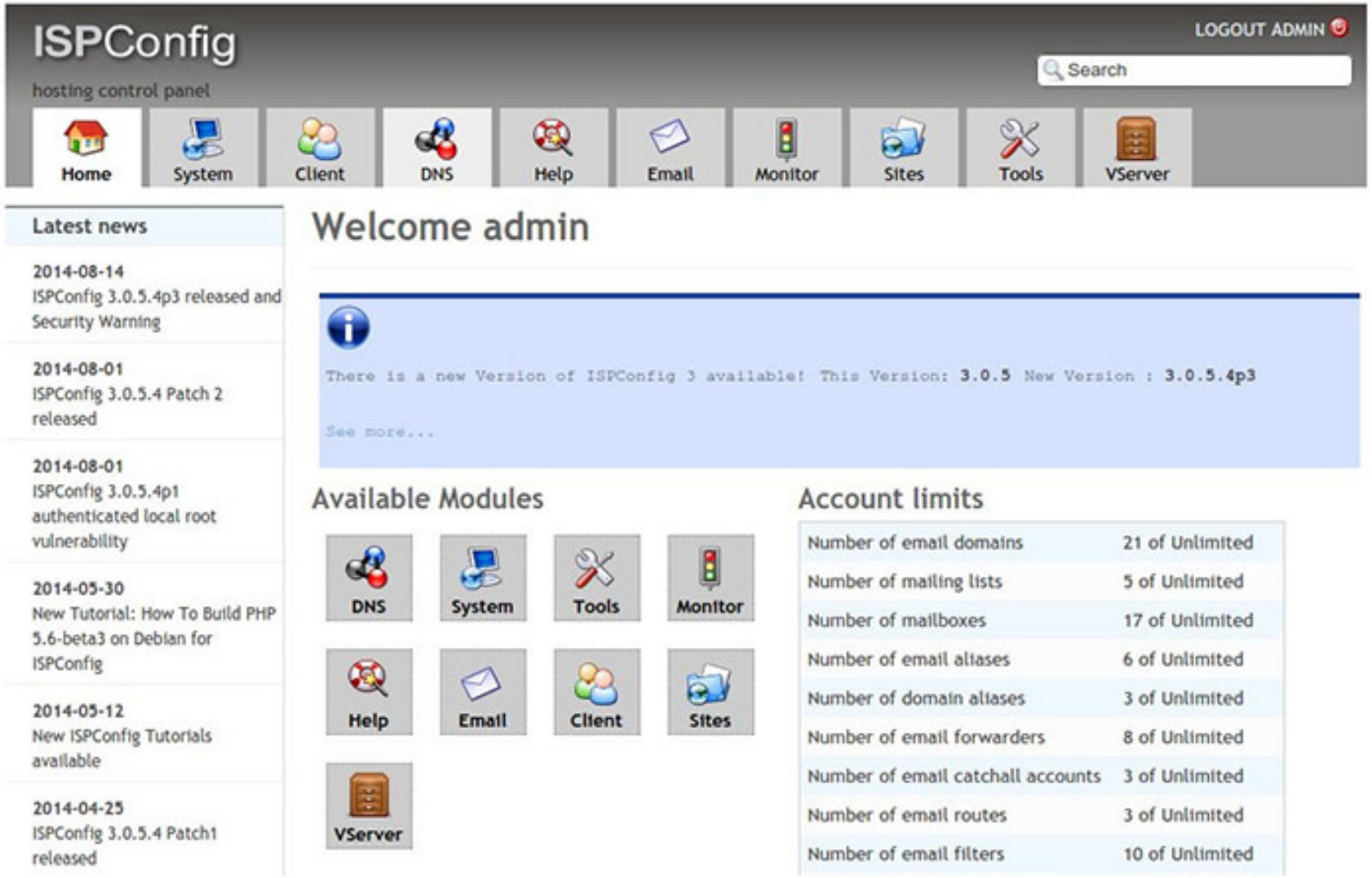
- Der er en lang række web kontrol paneler, men vi vil fokusere på den meget populære cPanel/WHM og den open source ISPconfig.
- cPanel er det mest anvendte web kontrolpanel, cPanel er web kontrolpanel værktøj til webstedsejere og Web Host Manager (WHM) er server administrativt værktøj for hosting-udbydere.
- Både cPanel og WHM kan betragtes som de mest full-featured systemer der findes. Brugergrænsefladen, selvom den er let at bruge, er bestemt ikke den bedste blandt web kontrolpaneler.

Webserver Kontrolpaneler

Webserver Kontrolpaneler

- ISPConfig er et populært, open source web kontrolpanel system med god virksomheds støtte.
- Projektet hævder at have mere end 40.000 downloads om måneden.
- Der er god multi-server, IPv6, og virtualiserings (OpenVZ) støtte, som er ideel til internet-udbydere eller andre virksomhedsmiljøer.

Webserver Kontrolpaneler



Webserver ISPconfig

- ISPconfig er et open source hosting kontrolpanel til Linux der primært laves af ISPConfig UG, men siden det benytter BSD licens kan alle kommentere og rette i koden, hvilket sikrer at fejl og problemer hurtigt dokumenteres og rettes.
- ISPConfig giver mulighed for at administrere hjemmesider, e-mail-adresser, databaser og DNS optegnelser via et web-baseret interface.
- Der er fire 4 login niveauer til systemet som standard: administrator, forhandler, klient og email-bruger.

Programmering af tråde

Herunder multithreading

Eksempler i C#

Programmering af tråde

- En tråd er defineret som udførelsesvejen af et program.
- Hver tråd definerer en unik strøm af kontrol.
  - Hvis en applikation indebærer komplicerede og tidskrævende operationer, så er det ofte nyttigt at indstille forskellige udførelses stier eller tråde, hvor hver tråd udfører et bestemt job.
- En tråd kaldes også en letvægts proces (lightweight process).
- Et almindeligt eksempel på brug af tråde er gennem-førelsen af parallel programmering af moderne operativsystemer.
- Anvendelse af tråde sparer spild af CPU cyklus og øge effektiviteten af en applikation.

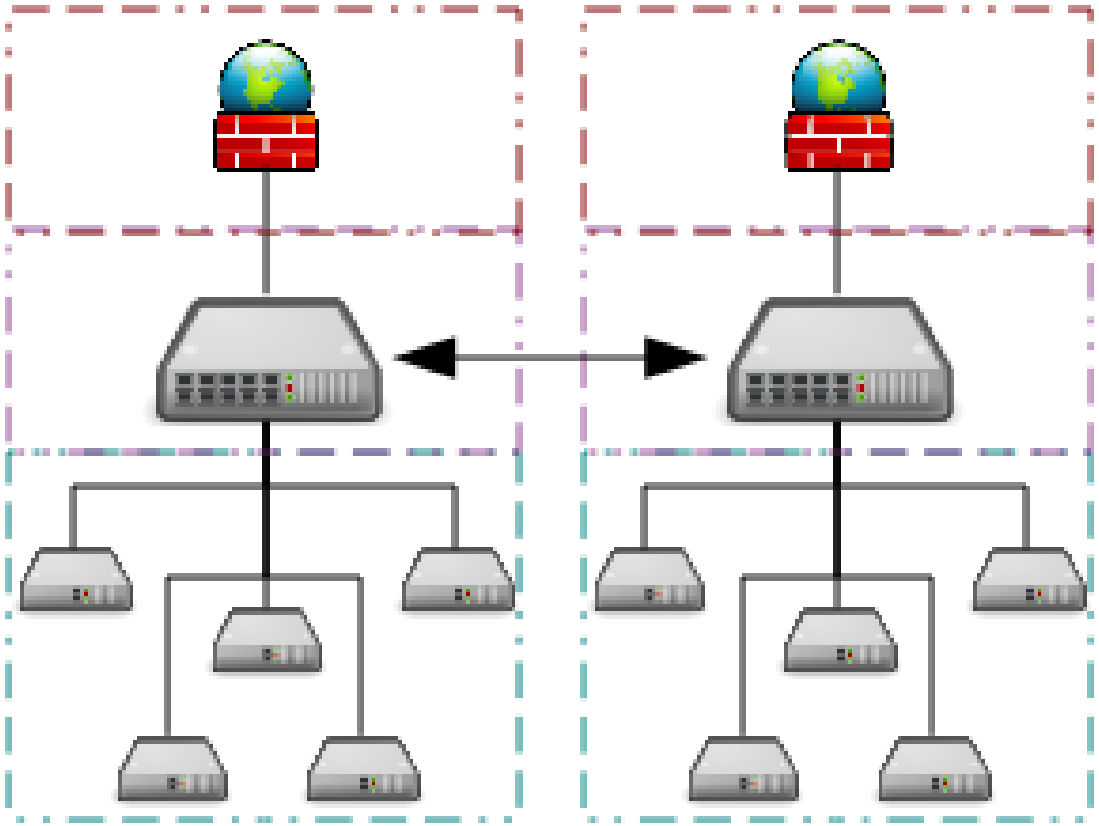
Programmering af tråde
En tråds livs cyklus

- En tråd er defineret som udførelsesvejen af et program.
- Hver tråd definerer en unik strøm af kontrol.
  - Hvis en applikation indebærer komplicerede og tidskrævende operationer, så er det ofte nyttigt at indstille forskellige udførelses stier eller tråde, hvor hver tråd udfører et bestemt job.
- En tråd kaldes også en letvægts proces (lightweight process).
- Et almindeligt eksempel på brug af tråde er gennem-førelsen af parallel programmering af moderne operativsystemer.
- Anvendelse af tråde sparer spild af CPU cyklus og øge effektiviteten af en applikation.

Programmering af tråde Inter-process communication

- Inter-proces kommunikation eller InterProcess CommuniCation (IPC) henviser specifikt til de mekanismer som et operativsystem tilbyder til at lade de processer, det administrerer, dele data.
- Typisk vil programmer, der benytter IPC, blive kategoriseret som klienter og servere, hvor klienten anmoder om oplysninger og serveren reagerer på anmodninger fra klienterne.
- Mange programmer er både klienter og servere, som almindeligvis ses i distribueret databehandling.
- En IPC kan være mange ting, fra remote procedure kald over operativ systems kommunikations stakken til distribuerede objekt modeller.

Programmering af tråde Inter-process communication



- Et eksempel, der viser grid computing, der forbinder mange personlige computere over internettet ved hjælp af inter-proces netværkskommunikation

Programmering af tråde Inter-process communication

Former for inter-proces kommunikation

- Filer benyttes af de fleste operativsystemer, enten på en disk eller på en fil-server, og de kan tilgås af flere processer.
- Signal er igen noget de fleste operativsystemer benytter når en system besked bliver sendt fra proces til proces. Anvendes normalt ikke til at overføre data, men i stedet bruges til at fjernstyre partner processen.
- Socket er også en alment anvendt teknik, hvor en datastrøm sendes over et netværksinterface, enten til en anden proces på den samme computer eller til en anden computer på netværket. En socket respekterer *ikke* beskedens grænser og de skal derfor formatteres.

Programmering af tråde Inter-process communication

Former for inter-proces kommunikation

- Message queue er også noget de fleste OS benytter. Det er en datastrøm der minder om en socket, men som regel bevarer beskedens grænser. Typisk er de implementeret i operativsystemet og tillader flere processer at læse og skrive til message queue uden at de er direkte forbundet med hinanden.
- Pipe er en ensrettet datakanal der benyttes af alle POSIX operativ systemer og Windows. Data skrevet til skrive enden af røret bliver bufferet af operativsystemet indtil den læses fra læse enden af røret. To-vejs datastrømme mellem processer kan opnås ved at skabe to rør der udnytter standard input og output.

Programmering af tråde Inter-process communication

Former for inter-proces kommunikation

- Named Pipe bruge ligesom pipe af POSIX og Windows systemer. En pipe gennemføres via en fil på filsystemet i stedet for standard input og output. Flere processer kan læse og skrive til filen som en buffer for IPC-data.
- Semaforens er igen POSIX og Windows systemer. En semafor er en enkel struktur, der synkroniserer flere processer, der handler på fælles ressourcer. Vi kigger nærmere på dem om lidt.
- Delt hukommelse er igen POSIX og Windows systemer. Flere processer får adgang til den samme blok af hukommelsen, hvilket skaber en fælles buffer for processerne til at kommunikere med hinanden.

Programmering af tråde Inter-process communication

Former for inter-proces kommunikation

- Memory mapped er igen POSIX og Windows systemer. En fil mapped til RAM som kan modificeres ved at ændre hukommelsesadresser direkte i stedet for at sende til en strøm. Dette deler de samme fordele som en standard fil.

Programmering af tråde Semaforens

- I datalogi er en semaforens en variabel eller abstrakt datatype, der bruges til at kontrollere adgang til en fælles ressource af flere processer i et samtidig (concurrent) system, såsom et multiprogram operativ-system.
- En triviel semaforens er en almindelig variabel, der ændres (fx inkrementeres eller dekrementeres, eller toggles) afhængig af programmerings definerede betingelser. Variablen anvendes derefter som en betingelse for at styre adgangen til nogle system ressourcer.

Programmering af tråde Semaforens

Antag et bibliotek har 10 identiske studie rum, der skal bruges af en elev ad gangen. For at forebygge stridigheder skal de studerende anmode om et rum i receptionen, hvis de ønsker at gøre brug af et studie rum. Hvis ingen værelser er ledige må de studerende vente ved skrivebordet indtil nogen giver afkald på et rum. Når en studerende er færdig med at bruge et rum, skal den studerende vende tilbage til skrivebordet og indikere, at det ene rum er blevet ledigt.

I den simpleste implementering, er den bibliotekaren i receptionen ikke nødt til at holde styr på, hvilke rum der er besat, eller hvem der bruger dem, og hun ved heller ikke om et givent rum faktisk bliver brugt, kun antallet af ledige lokaler til rådighed. Det kender hun også kun korrekt, hvis alle de studerende faktisk bruger deres rum, mens de har tilmeldt sig dem og returnerer dem, når de er færdige. Når en studerende anmoder om et rum formindsker bibliotekaren dette tal. Når en studerende frigiver et rum, forøger bibliotekaren dette tal. Når adgangen til et rum er givet, kan rummet bruges så længe som ønsket, og dermed er det er ikke muligt at bestille værelser i god tid.

I dette scenario repræsenterer skrivebords tælleren en optællings semaforens, rummene er ressourcerne og de studerende repræsenterer processer. Værdien af semaforens i dette scenario er til at begynde med 10. Når en studerende anmoder om en plads, får han eller hun adgang, og værdien af semaforens ændres til 9. Efter den næste elev kommer falder det til 8, derefter 7 og så videre. Hvis nogen anmoder et rum, og den resulterende værdi semaforens vil være negativ er de tvunget til at vente indtil et rum frigøres (når optællingen forøges fra 0).

Programmering af tråde Semaforens

- I datalogi er en semaforens en variabel eller abstrakt datatype, der bruges til at kontrollere adgang til en fælles ressource af flere processer i et samtidig (concurrent) system, såsom et multiprogram operativ-system.
- En triviel semaforens er en almindelig variabel, der ændres (fx inkrementeres eller dekrementeres, eller toggles) afhængig af programmerings definerede betingelser. Variablen anvendes derefter som en betingelse for at styre adgangen til nogle system ressourcer.

Programmering af tråde Semaforens

- Når det bruges til at kontrollere adgangen til en pulje af ressourcer holder en semaforens kun øje med hvor mange ressourcer er ledige, den holder ikke styr på, *hvilke* af de ressourcer der er ledige. Andre mekanismer (muligvis involverende flere semaforer) kan være nødvendige for at vælge en bestemt ledig ressource.
- Paradigmet er særligt effektivt fordi semaforens tæller kan tjene som en nyttig udløser for en række forskellige handlinger.
  - Bibliotekaren tidligere kan slukke lysene i læse rummene, når der er ingen studerende er tilbage, eller placere et skilt, der siger rummene er meget travle, når de fleste af værelserne er besat.

Programmering af tråde Semaforens

- Succesen af protokollen kræver at applikationer følger den korrekt. Fairness og sikkerhed vil sandsynligvis blive kompromitteret (som praktisk talt betyder et program kan opføre sig langsomt, handle ujævnt, hænge eller gå ned), hvis blot en enkelt proces handler forkert. Dette omfatter:
  - Anmode om en ressource og glemme at frigøre den.
  - Frigive en ressource, der aldrig blev anmodet om.
  - Holde en ressource i lang tid uden behov for den.
  - Benytte en ressource uden at anmode om den først (eller efter at frigivet den).
- Selv hvis alle processer følger disse regler, kan multi-ressource *deadlocks* stadig forekomme, når der er forskellige ressurser, der forvaltes af forskellige semaforer og når processer skal bruge mere end én ressource ad gangen.

Programmering af tråde Semaforens

- Succesen af protokollen kræver at applikationer følger den korrekt. Fairness og sikkerhed vil sandsynligvis blive kompromitteret (som praktisk talt betyder et program kan opføre sig langsomt, handle ujævnt, hænge eller gå ned), hvis blot en enkelt proces handler forkert. Dette omfatter:
  - Anmode om en ressource og glemme at frigøre den.
  - Frigive en ressource, der aldrig blev anmodet om.
  - Holde en ressource i lang tid uden behov for den.
  - Benytte en ressource uden at anmode om den først (eller efter at frigivet den).
- Selv hvis alle processer følger disse regler, kan multi-ressource *deadlocks* stadig forekomme, når der er forskellige ressurser, der forvaltes af forskellige semaforer og når processer skal bruge mere end én ressource ad gangen.

Programmering af tråde Semaforens

- Counting semaforer er udstyret med to operationer, historisk betegnet som P og V.
  - Operation V forøger Semaforens S.
  - Operation P formindsker den
- Værdien af Semaforens S er antallet af enheder af den ressource, der er tilgængelige i øjeblikket.
- P operationen spilder tid eller sover indtil en ressource beskyttet af semaforens bliver tilgængelig, på hvilket tidspunkt ressourcen bliver krævet med det samme.
- V operation er den inverse: det stiller en ressource til rådighed igen efter processen er færdig med at bruge den.
- En vigtig egenskab ved Semaforens S er, at dens værdi ikke kan ændres, undtagen ved hjælp af V og P operationer.

Programmering af tråde Semaforens

- En enkel måde at forstå ventetid (P) og signal (V) operationer er:
  - vent (wait): Hvis værdien af semaforens variablen ikke er negativ, dekrementeres den med 1. Hvis Semaforens variabel er nu negativ, er proces udførelses Vent blokeret (dvs. sættes til semaforens kø), indtil værdien er større eller lig med 1. Hvis ikke fortsætter processens udførelse, havende anvendt en enhed af ressourcen.
  - signal: Forøger værdien af semaforens variablen med 1. Efter tilvækst, hvis præ-tilvækst værdi var negativ (hvilket betyder at der er processer der venter på en ressource), overføres en blokeret processen fra semaforens venter kø til klar kø.

Programmering af tråde Semaforens

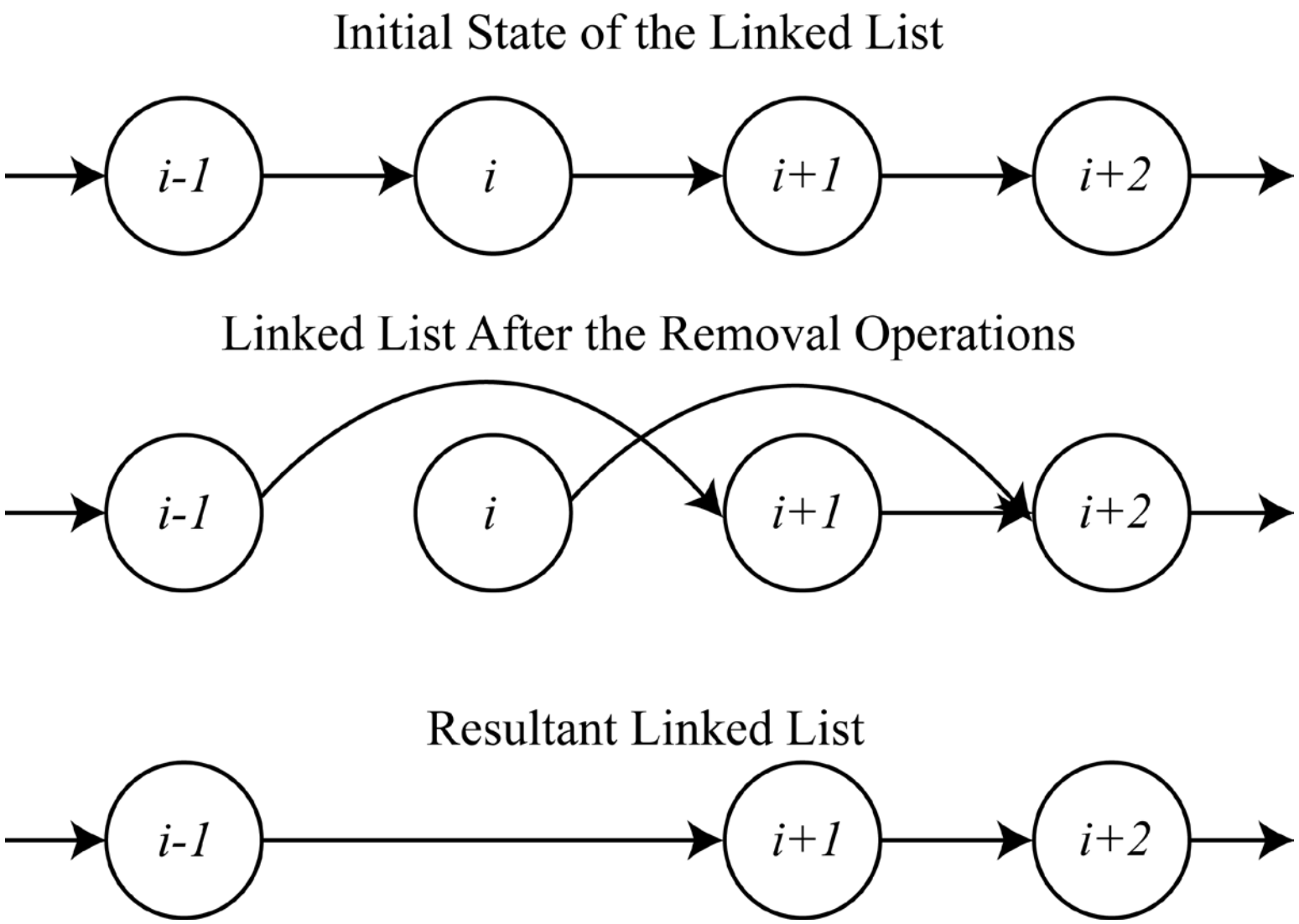
- Mange operativsystemer leverer effektive semaforens primitiver, der fjerner blokeringen af en ventende proces, når semaforens øges.
  Det betyder, at processerne ikke spilder tid på at kontrollere semaforens værdi unødigt.

Programmering af tråde
Mutual exclusion

- I datalogi er gensidig udelukkelse (mutual exclusion) en egenskab ved concurrency kontrol, som er lavet med henblik på at forebygge race condition / hazard.
  - Race condition er når output er afhængig af sekvensen eller timingen af andre ukontrollerbare begivenheder.
- Mutual exclusion er kravet om, at en udførelses-tråd aldrig når sin kritiske strækning på samme tid som en anden samtidig tråd udførelse går ind i sin egen kritiske sektion.

Programmering af tråde

Mutual exclusion



To nodes, *i* og *i* + 1, fjernes samtidigt hvilket resulterer i at node *i* + 1 ikke fjernes.

Programmering af tråde
Dekkers algoritme

- Dekker algoritme er det første kendte rigtige løsning til den gensidige udelukkelses problem i parallel programmering.
- Løsningen tilskrives den hollandske matematiker Th. J. Dekker af Edsger Dijkstra, der var en af de definerende programmører fra 1950'erne og frem.
- Den lader to tråde dele en enkelt ressource uden konflikt ved kun at benytte delt hukommelse til kommunikation.
- Den undgår den strenge vekslen af en naiv tur-skift algoritme og var en af de første gensidig udelukkelses algoritmer, der blev opfundet.

Programmering af tråde
Dekkers algoritme

- Hvis to processer forsøger at gå i en kritisk sektion på samme tid, vil algoritmen kun tillade én proces, baseret på hvis tur det er.
- Hvis en proces er allerede i den kritiske sektion vil den anden proces travlt vente (busy wait) på at den første proces afsluttes. Dette gøres ved brug af to flag, wants_to_enter [0] og wants_to_enter [1], der indikerer en hensigt om at komme ind på den kritiske strækning for processerne 0 og 1 respektivt, og en variabel tur, der angiver, hvem der har prioritet mellem de to processer.

Programmering af tråde
Dekkers algoritme

Dekkers algoritme kan udtrykkes i følgende pseudokode:

```
variables
    wants_to_enter : array of 2 booleans
    turn : integer

wants_to_enter[0] ← false
wants_to_enter[1] ← false
turn ← 0   // or 1
```

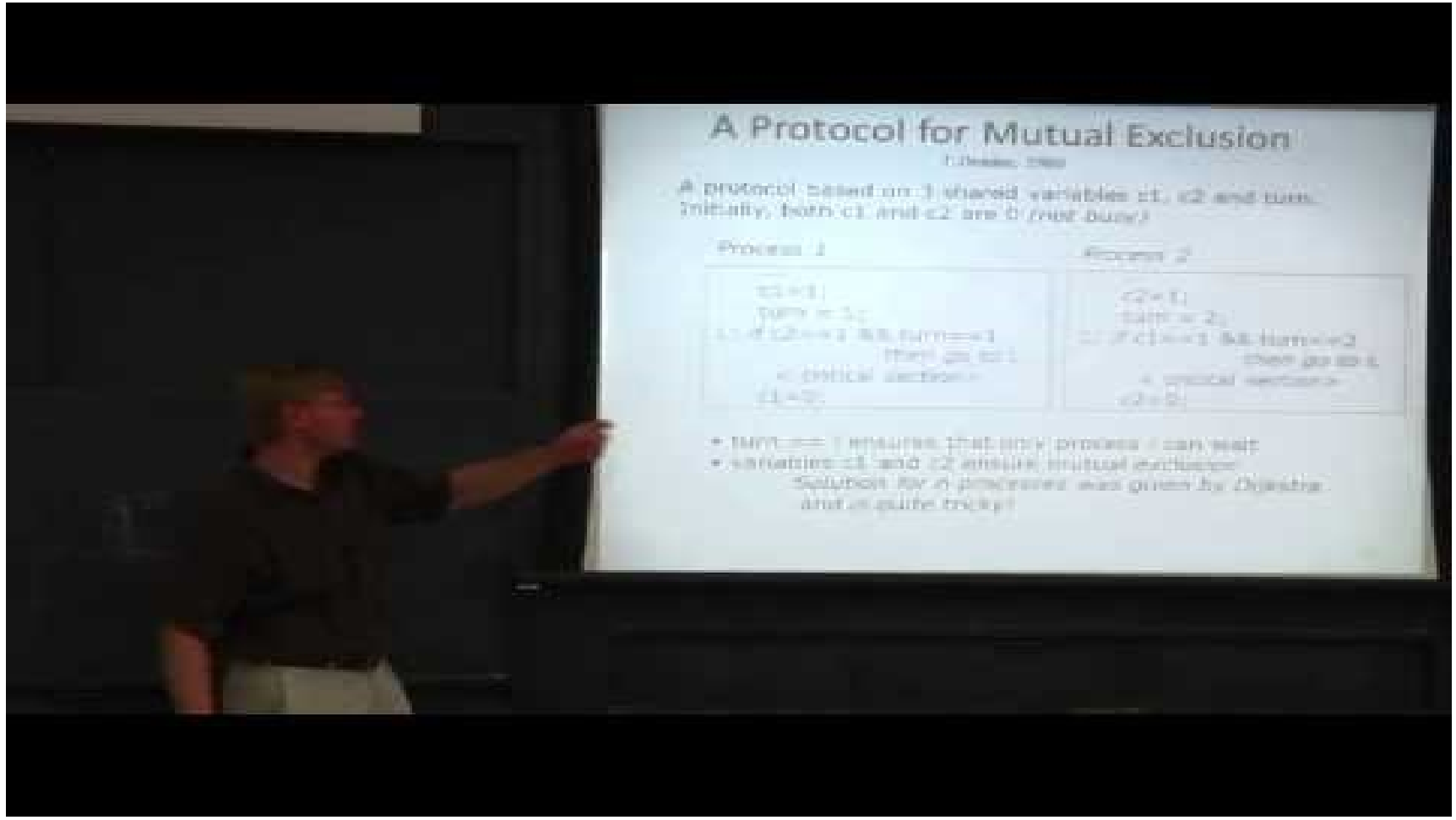
```
p0:
   wants_to_enter[0] ← true
   while wants_to_enter[1] {
      if turn ≠ 0 {
         wants_to_enter[0] ← false
         while turn ≠ 0 {
           // busy wait
         }
         wants_to_enter[0] ← true
      }
   }

   // critical section
   ...
   turn ← 1
   wants_to_enter[0] ← false
   // remainder section
```

```
p1:
   wants_to_enter[1] ← true
   while wants_to_enter[0] {
      if turn ≠ 1 {
         wants_to_enter[1] ← false
         while turn ≠ 1 {
           // busy wait
         }
         wants_to_enter[1] ← true
      }
   }

   // critical section
   ...
   turn ← 0
   wants_to_enter[1] ← false
   // remainder section
```

Programmering af tråde



Fil system

At gemme filer på disken

Fil systemet

- En fil er en samling gemt på en disk med et specifikt navn og mappe sti. Når filen åbnes til læsning eller skrivning bliver den en stream.
- En stream er grundlæggende skevensen af bytes der går gennem kommunikations stien. Der er to hoved streams: **input stream** og the output stream.
  - Input stream bruges til at læse data fra filen (read operation)
  - og output stream bruges til at skrive til filen (write operation).

Fil systemet I/O klasser

- System.IO namespace har en række klasser der bruges til at udføre en vifte af operationer med filer
  - Lave filer
  - Slette filer
  - Læse fra eller skrive til en fil
  - Lukke en fil osv.

I/O Class	Description
BinaryReader	Reads primitive data from a binary stream.
BinaryWriter	Writes primitive data in binary format.
BufferedStream	A temporary storage for a stream of bytes.
Directory	Helps in manipulating a directory structure.
DirectoryInfo	Used for performing operations on directories.
DriveInfo	Provides information for the drives.
File	Helps in manipulating files.
FileInfo	Used for performing operations on files.
FileStream	Used to read from and write to any location in a file.
MemoryStream	Used for random access to streamed data stored in memory.
Path	Performs operations on path information.
StreamReader	Used for reading characters from a byte stream.
StreamWriter	Is used for writing characters to a stream.
StringReader	Is used for reading from a string buffer.
StringWriter	Is used for writing into a string buffer.

Fil systemet
FileStream klassen

- FileStream klassen i System.IO namespace hjælper med at skrive fra, skrive til og lukke filer. Klassen er afledt af den abstrakte klasse Stream.
- For at lave en ny fil eller åbne en eksisterende fil skal man have et **FileStream** objekt.

```
FileStream <object_name> = new FileStream( <file_name>,
<FileMode Enumerator>, <FileAccess Enumerator>,
<FileShare Enumerator>);
```

- Her laves et FileStream objekt F til at læse filen sample.txt

```
FileStream F = new FileStream("sample.txt", FileMode.Open,
FileAccess.Read, FileShare.Read);
```

Parameter	Description
FileMode	The **FileMode** enumerator defines various methods for opening files. The members of the FileMode enumerator are: ▫ **Append**: It opens an existing file and puts cursor at the end of file, or creates the file, if the file does not exist. ▫ **Create**: It creates a new file. ▫ **CreateNew**: It specifies to the operating system, that it should create a new file. ▫ **Open**: It opens an existing file. ▫ **OpenOrCreate**: It specifies to the operating system that it should open a file if it exists, otherwise it should create a new file. ▫ **Truncate**: It opens an existing file and truncates its size to zero bytes.
FileAccess	**FileAccess** enumerators have members: **Read**, **ReadWrite** and **Write**.
FileShare	**FileShare** enumerators have the following members: ▫ **Inheritable**: It allows a file handle to pass inheritance to the child processes ▫ **None**: It declines sharing of the current file ▫ **Read**: It allows opening the file for reading ▫ **ReadWrite**: It allows opening the file for reading and writing ▫ **Write**: It allows opening the file for writing

Fil systemet
FileStream klassen

```
1   using System;
2   using System.IO;
3
4   namespace FileIOApplication
5   {
        0 references
6       class Program
7       {
            0 references
8           static void Main(string[] args)
9           {
10              FileStream F = new FileStream("test.dat", FileMode.OpenOrCreate, FileAccess.ReadWrite);
11              for (int i = 1; i <= 20; i++)
12              {
13                  F.WriteByte((byte)i);
14              }
15
16              F.Position = 0;
17              for (int i = 0; i <= 20; i++)
18              {
19                  Console.Write(F.ReadByte() + " ");
20              }
21              F.Close();
22              Console.ReadKey();
23          }
24      }
25  }
```

Fil systemet

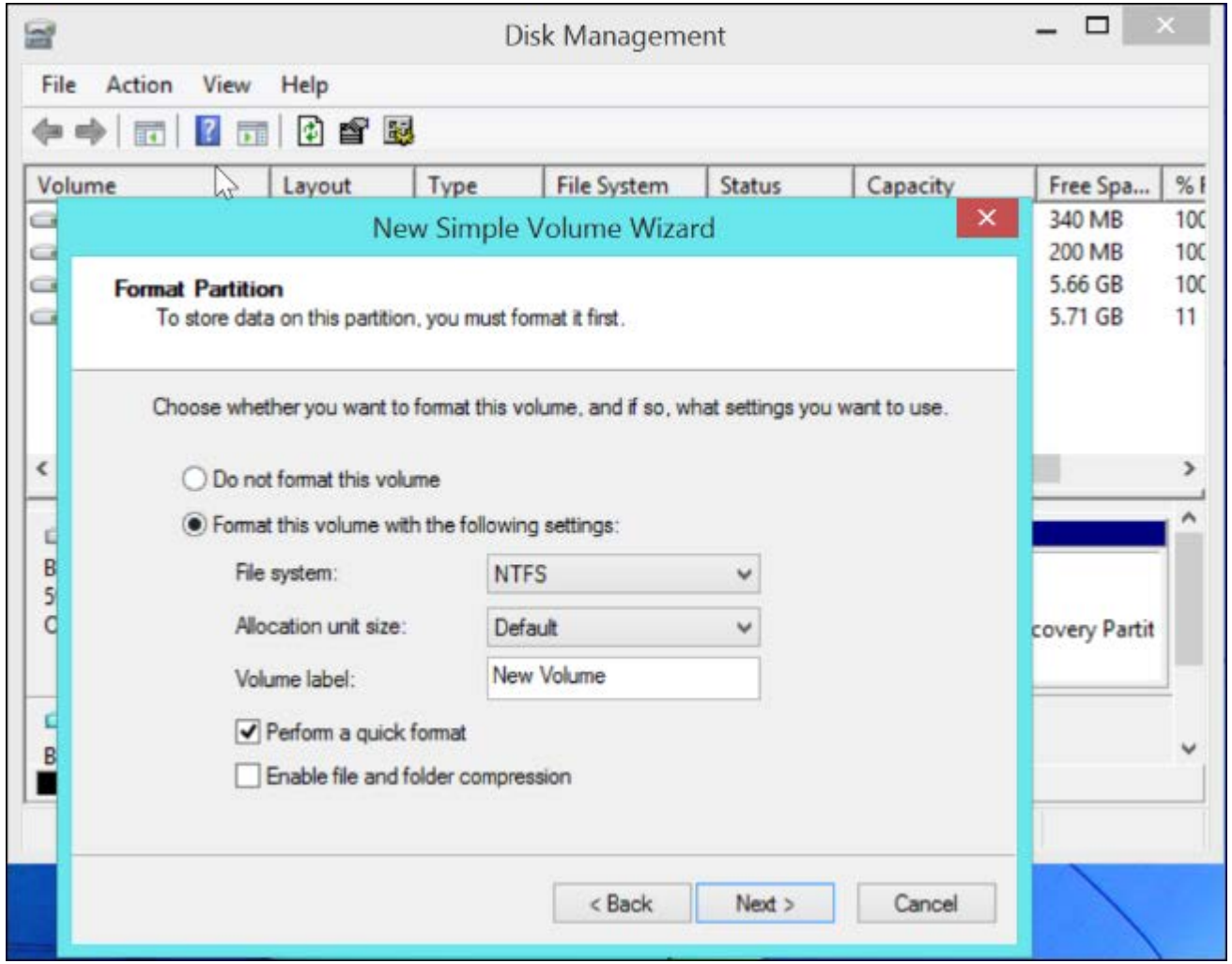
- Forskellige filsystemer er ”bare” forskellige måder at organisere og lagre filer på en harddisk, flash-drev, eller en anden lagerenhed.
- Hver lagerenhed har en eller flere partitioner, og hver partition der er "formateret" med et filsystem.
- Formaterings processen skaber blot et tomt filsystem af denne type på enheden.
- Et filsystem sørger for en måde at adskille data på drevet i individuelle stykker, som er filerne. Det giver også en måde at gemme data om disse filer - fx deres filnavne, tilladelser og andre attributter.
- Filsystemet har også et indeks - en liste over de filer på drevet, og hvor de er placeret på drevet, så styresystemet kan se, hvad der er på drevet på ét sted i stedet for gennemgå hele drevet for at finde en fil .

Fil systemet

- Et operativsystemet er nødt til at forstå et filsystem, så det kan vise dets indhold, åbne filer, og gemme filer til det.
  - Hvis operativsystemet ikke forstår et filsystem, kan man være i stand til at installere en filsystem driver, der sørger for understøttelse – ellers kan man ikke bruge filsystemet sammen med at operativsystemet.
- Der er ikke noget bedst filsystem til alle brugsmønstre.
- Hvert operativsystem har tendens til at bruge sit egen fil-system, som operativsystemets udviklere også arbejde på.
  - Microsoft, Apple og Linux-kernens udviklere har alle arbejde på tderes egne filsystemer.
- Nye filsystemer kan være hurtigere, mere stabile, skalere bedre til større lagringsenheder, og have flere funktioner end gamle.

Fil systemet

- Et filsystem er ikke som en partition, der groft sagt er en klump lagerplads. Et filsystem specificerer hvordan filer bliver lagt ud, organiseret, indekseret, og hvordan metadata er forbundet med dem.



Fil systemet

Forskellige filsystemer

Dette er nogle af de mest udbredte filsystemer

- FAT32: FAT32 er et ældre Windows-filsystem, men det er stadig brugt på flytbare medieenheder - dog kun de mindre. Større eksterne harddiske på 1 TB eller deromkring vil sandsynligvis komme formateret med NTFS. Man vil kun bruge dette med små lagerenheder eller for kompatibilitet med andre enheder som digitale kameraer, spilkonsoller, set-top-bokse, og andre enheder, der kun understøtter FAT32 og ikke det nyere NTFS-filsystem.
  Man kan dog højest have filer på 2GB og filnavne på 255 tegn.
  Diske mellem 512MB og 8TB understøttes.
  FAT32 gemmer hverken information om ejeren af filen, POSIX fil rettigheder, meta-data ændringer eller checksums.

Fil systemet

Forskellige filsystemer

- NTFS: Moderne versioner af Windows - siden Windows XP - bruger NTFS-filsystemet til deres system partition. Eksterne drev kan formateres med enten FAT32 eller NTFS.
  Både filer og diske kan være op til 16 EB store.
  NTFS understøtter information om ejeren af filen, POSIX fil rettigheder, meta-data ændringer og checksums.

Fil systemet

Forskellige filsystemer

- HFS+: Macs bruger HFS+ til deres interne partitioner, og de foretrækker også at formatere eksterne drev med HFS+ - det er nødvendigt at bruge et eksternt drev med Time Machine, så filsystemet attributter kan blive ordentligt sikkerhedskopieret, for eksempel.
  Macs kan også læse og skrive til FAT32 filsystemer, selv om de kun kan læse fra NTFS filsystemer som standard - du behøver tredjeparts software til at skrive til NTFS filsystemer fra en Mac.
  Både filer og diske kan være op til en smule under 8 EB.
  HFS+ understøtter information om ejeren af filen, POSIX fil rettigheder, meta-data ændringer og checksums

Fil systemet

Forskellige filsystemer

- Ext2 / Ext3 / EXT4: Du vil ofte se Ext2, Ext3, og EXT4 filsystemer på Linux.
- Ext2 er et ældre filsystemer, og det mangler vigtige funktioner som journalisering - hvis strømmen går ud eller en computer går ned, mens du skriver til et ext2-drev, kan data gå tabt.
- Ext3 tilføjer disse robustheds funktioner på bekostning af en vis hastighed.
- Ext4 er mere moderne og hurtigere - det er standard filsystemet på de fleste Linux-distributioner nu, og er hurtigere.
- Windows og Mac understøtter ikke disse filsystemer - du skal bruge et tredjeparts værktøj til at få adgang til filer på sådanne filsystemer.
- Af denne grund, er det ofte ideel til at formatere dine Linux-system partitioner som ext4 og efterlade flytbare enheder formateret med FAT32 eller NTFS, hvis du har brug for kompatibilitet med andre operativsystemer. Linux kan læse og skrive til både FAT32 eller NTFS.

Fil systemet
Forskellige filsystemer

- ZFS: ZFS er et kombineret filsystem og logisk volumen manager designet af Sun Microsystems. Funktionerne i ZFS omfatter beskyttelse mod data korruption, støtte til høj lagerkapacitet, effektiv datakomprimering, integrering af begreberne filsystem og volumen kontrol, snapshots og copy-on-write-kloner, kontinuerlig integritets kontrol og automatisk reparation, RAID-Z og native NFSv4 ACL.
  Filer kan være op til 16EB og diske helt op til 256ZB understøttes.
  ZFS understøtter information om ejeren af filen, POSIX fil rettigheder, meta-data ændringer, checksums og mere til.

BSD

Et ægte Unix operativsystem

BSD

- I open source verdenen er ordet "Linux" næsten synonymt med "Operativ System", men det er ikke det eneste open source UNIX® operativsystem.
- BSD står for "Berkeley Software Distribution".
- Det er navnet på distributioner af kildekoden fra University of California, Berkeley, som oprindeligt var udvidelser til AT&T Research UNIX® operativsystemet.
- Flere open source operativsystem projekter er baseret på en frigivelse af denne kildekode kendt som 4.4BSD-Lite.
- Desuden udgør de et antal pakker fra andre Open Source projekter, herunder især GNU-projektet.

BSD

- Det samlede styresystem omfatter:
  - BSD kernel, der håndterer process scheduling, memory management, symmetric multi-processing (SMP), device drivers, etc.
  - C biblioteket, den grundlæggende API til systemet.
    - BSD C biblioteket er baseret på kode fra Berkeley, ikke GNU projektet.
  - Utilities som shells, file utilities, compilers og linkers.
    - Nogle af disse utilities er afledt af GNU projektet, andre er ikke.
  - X Window system, der håndterer grafisk display.
    - X Window systemet brugt i de fleste versioner af BSD bliver vedligeholdt af X.Org projektet.
    - FreeBSD tillader brugeren at vælge mellem en vifte af desktop environments, som Gnome, KDE eller Xfce; og lightweight window managers som Openbox, Fluxbox eller Awesome.
  - Mange andre programmer og utilities.

BSD

- BSD operativsystemer er ikke kloner, men open source derivater af AT&T Research UNIX® operativsystem, som også er stamfader til den moderne UNIX® System V.
- Noget af BSD koden blev udgiver som open source (uden AT&Ts ejede dele) i 1990, men 20% manglede.
- De blev føjet til i 1992 og det hed samlet 386BSD, men den var ikke stabil.
- Den ledte dog i 1993 til de to projekter NetBSD og FreeBSD, der kører i dag.
- I 1996 splittede OpenBSD af fra NetBSD og i 2003 DragonFlyBSD fra FreeBSD.

BSD
MacOS

- MacOS (oprindeligt kaldt "Mac OS X" indtil 2012 og derefter "OS X" indtil 2016) er det nuværende Mac-operativsystemet, der officielt afløste det klassiske Mac OS i 2001.
- Selv om systemet oprindeligt blev markedsført som blot "udgave 10" af Mac OS har det en historie, der er stort set uafhængig af det klassiske Mac OS.
- Det er et Unix-baseret operativsystem bygget på NeXTStep og anden teknologi udviklet af NeXT fra slutningen af 1980'erne til begyndelsen af 1997, da Apple købte selskabet og selskabets administrerende direktør Steve Jobs vendte tilbage til Apple
- MacOS gør brug af BSD kodebase (primært afledt af OpenBSD) og XNU kernen, og dens centrale sæt af komponenter er baseret på Apples open source Darwin operativsystem.

BSD
MacOS


Finder File Edit View Go Window Help
Fri 6:15 PM
About This Mac
Get Mac OS X Software...
System Preferences...
Dock
Location
Recent Items
Force Quit...
Sleep
Restart
Shut Down
Log Out...
Turn Magnification On
Turn Hiding On
Dock Preferences...
Welcome to Mac OS X
Mac OS X Install CD
midnite
Back
View
Computer
Home
Favorites
Applications
Desktop
Documents
Library
Movies
Music
Pictures
Public
Sites
Show All
Displays
Sound
Network
Startup Disk
Appearance:
Blue
Highlight color:
Graphite
Click in the scroll bar to:
Jump to next page
Scroll to here

BSD
FreeBSD


How-to install FreeBSD 10.2
Plus the Gnome desktop and basic applications
On real
hardware

NAS

Filserver på netværket

NAS

- Netværk-attached storage (NAS) er en type af dedikeret fil lagringsenhed, der giver lokalnetværk lokale netværk (LAN) nodes med filbaseret delt storage via en standard Ethernet-forbindelse.
- NAS-enheder, som typisk ikke har et tastatur eller skærm, konfigureres og styres via et browserbaseret hjælpeprogram. Hvert NAS bor på LAN som uafhængigt netværksknudepunkt og har sin egen IP-adresse.
- En vigtig fordel ved NAS er dens evne til at give flere klienter på netværket med adgang til de samme filer.

NAS

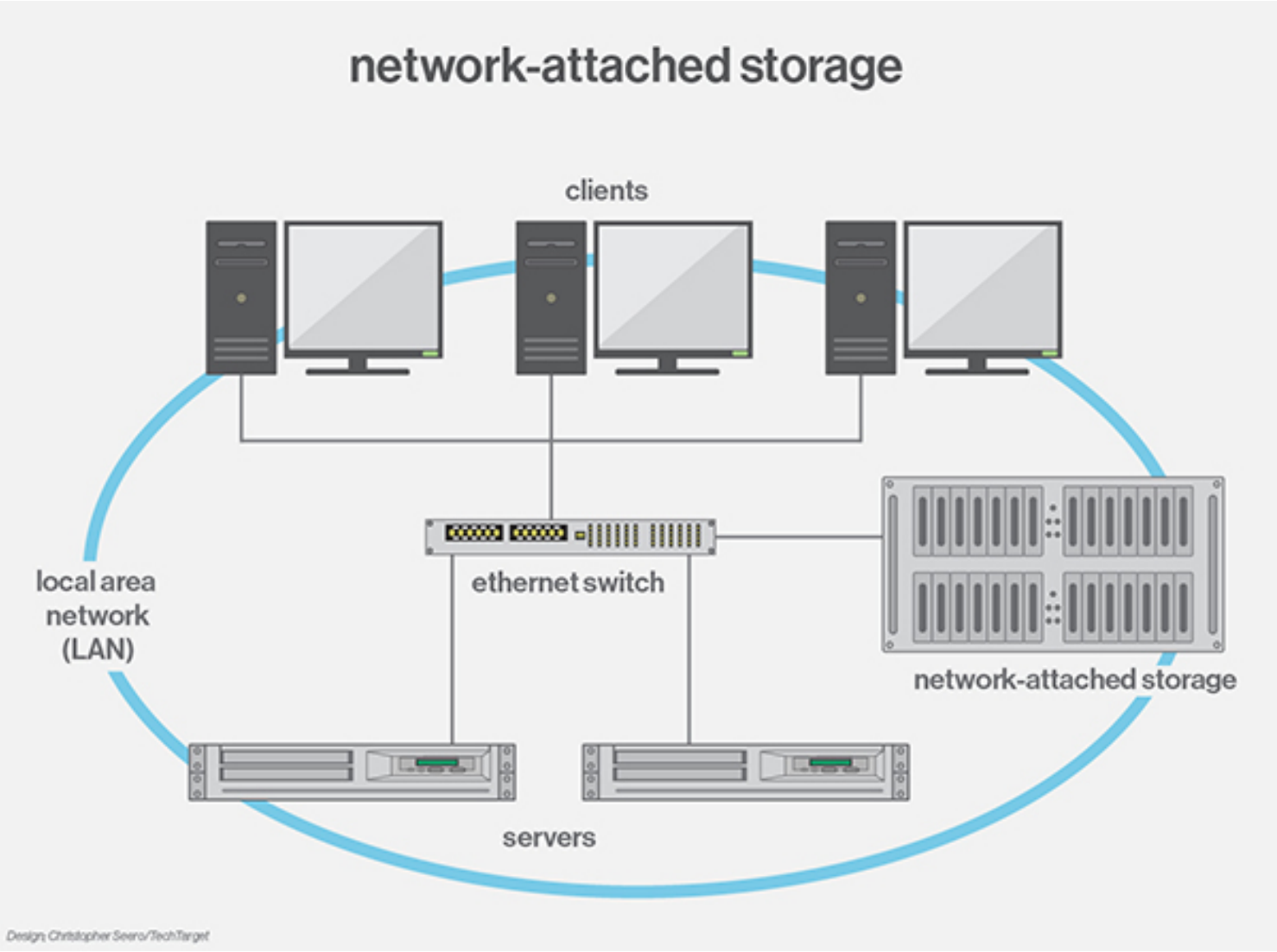
- Før NAS havde virksomheder typisk flere hundrede eller endda tusindvis af diskrete filservere, der skulle separat konfigureres og vedligeholdes.
  I dag, hvor mere lagerkapacitet er påkrævet kan NAS apparater simpelthen være udstyret med større diske eller grupperet sammen for at give både lodret skalerbarhed og vandret skalerbarhed.Mange NAS leverandører partner med cloud storage udbydere til at give kunderne et ekstra lag redundans til sikkerhedskopiering af filer.

NAS

Usecases

- I hjemmet bliver NASes ofte anvendt til opbevaring og serving af multimediefiler og til automatiseret backup.
- Mange intelligente hjem stoler på NAS til at yde central lagring for smart tv, sikkerhedssystemer og andre internet of things (IoT) komponenter i hjemmet.
- I virksomheden kan et NAS-array bruges som backup mål for arkivering og disaster recovery.
- Hvis en NAS-enhed har en server-tilstand kan det også fungere som en e-mail, multimedia, database eller printserver for en lille virksomhed.
- Nogle higher-end NAS-produkter kan holde nok diske til at understøtte RAID, en storage-teknologi, der forvandler flere harddiske til en logisk enhed, for at give bedre ydeevne, højere tilgængelighed og redundans.

NAS


network-attached storage
clients
local area network (LAN)
ethernet switch
network-attached storage
servers
Design: Christopher Severo/TechTarget

NAS versus DAS

- Direkte attached storage (DAS) er opbevaring på en dedikeret server eller lagerenhed, der ikke er på et netværk.
- For at få adgang til filer gemt på direkte-attached storage, skal slutbrugeren har fysisk adgang til enheden, hvor filerne er gemt.
- Fordelen ved DAS er, at det kan give slutbrugerne bedre ydeevne end NAS, hvilket er vigtigt for beregningsintensive softwareprogrammer.
- Ulempen ved DAS er, at det kræver opbevaring på hver enhed, der skal styres separat, hvilket kan komplicere den måde filer styres og deles.

NAS versus SAN

- Et storage-netværk (SAN) arrangerer opbevaring ressourcer på et uafhængigt, højtydende netværk.
- Den afgørende forskel mellem NAS og SAN er at network-attached storage håndterer input / output (I/O) ansøgninger om individuelle filer, mens et storage-netværk håndterer I/O-anmodninger om sammenhængende blokke af data.
- I dag kan nogle SAN transportere data over en standard Ethernet-forbindelse, men oftest bruger storage area networks Fibre Channel protokollen, som blev udviklet specielt til high-speed data transport på lager-area networks.

FreeNAS

En FreeBSD gren fokuseret på NAS brug

FreeNAS

Kilder

Materiale benyttet i denne lektion
Noget af det er udover pensum-listen!

Kilder

Webserver

- http://askubuntu.com/questions/52147/how-can-i-access-apache-on-virtualbox-guest-from-host
  (til dem af jer der har fulgt guiden om at sætte en lokal server op i Debian på en virtuel maskine og har problemer)
- http://whatis.techtarget.com/definition/Web-server
- https://youtu.be/wukg_d05AV4
- http://news.netcraft.com/archives/2016/04/21/april-2016-web-server-survey.html
- http://stackoverflow.com/questions/24338908/apache-or-nginx-i-like-to-understand-the-basic-working-flow-of-nginx-its-adv
- http://www.hostingadvice.com/blog/cpanel-vs-plesk-vs-webpanel/

Kilder

Programmering af tråde

- https://www.tutorialspoint.com/operating_system/os_processes.htm
- https://en.wikipedia.org/wiki/Inter-process_communication
- https://en.wikipedia.org/wiki/Semaphore_(programming)
- https://en.wikipedia.org/wiki/Dining_philosophers_problem
- http://web.mit.edu/6.005/www/fa14/classes/20-queues-locks/message-passing/
- https://en.wikipedia.org/wiki/Mutual_exclusion
- https://en.wikipedia.org/wiki/Race_condition
- https://en.wikipedia.org/wiki/Busy_waiting
- https://en.wikipedia.org/wiki/Dekker's_algorithm
- https://en.wikipedia.org/wiki/Pseudocode

Kilder

Fil system

- https://www.tutorialspoint.com/csharp/csharp_file_io.htm
- https://en.wikipedia.org/wiki/File_system
- https://en.wikipedia.org/wiki/Comparison_of_file_systems
- https://en.wikipedia.org/wiki/List_of_file_systems
- http://www.howtogeek.com/196051/htg-explains-what-is-a-file-system-and-why-are-there-so-many-of-them/
- Https://en.wikipedia.org/wiki/ZFS

BSD

- https://www.freebsd.org/doc/en/articles/explaining-bsd/
- https://en.wikipedia.org/wiki/Macintosh_operating_systems
- https://youtu.be/bVSXXeiFLgk
- https://youtu.be/bNuSTAKQDRU

Kilder

NAS

- http://searchstorage.techtarget.com/definition/network-attached-storage

FreeNAS

- https://ramsdenj.com/2016/01/01/freenas-server-build.html
- https://youtu.be/9zGrf88tty4