

---

# Operativsystem

Hösten 2013

[https://abacus.abo.fi/ro.nsf/W/opsys\\_13](https://abacus.abo.fi/ro.nsf/W/opsys_13)

# Innehåll

---

- Operativsystem, föreläsningar 26 h
- Laborationer 5 st, 10 h

# Föreläsningar

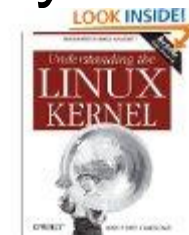
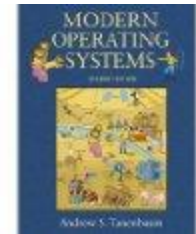
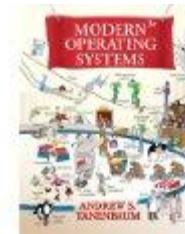
---

- Måndagar 15-17 (16.30), aud A3058
- Torsdagar 13-15 aud. Gamma
- Fredagar 8-10, aud A3058 / ESlabbet
- Kurstent 1: 20.12, Kurstent 2: 13.1
- Kom ihåg tentanmälning i MinPLAN

# Kursbok

---

- Andrew S. Tanenbaum:  
Modern Operating Systems, 2nd edition,  
Prentice Hall. 2007
  - Finns i kursboksbibban i 20 ex
- Annat intressant:
  - Stallings: Operating Systems, Internals and Design Principles
  - Deitel, Deitel & Choffnes: Operating Systems, Third Edition, Pearson Prentice Hall,
  - Daniel P. Bovet & Marco Cesati:  
Understanding the Linux Kernel



# Labbar / övningsuppgifter

---

- 5 st hålls på följande sätt
  - Hålls fredagar 10-12, Pingvin-farmen, eller Eslabbet (beroende på)
  - Praktisk del, där man jobbar med ett operativsystem (=Linux) och gör små programbitar / checkar egenskaper hos OS
  - Övningsuppgifterna inlämnas elektroniskt på <https://abacus.abo.fi/ro.nsf>
  - RÖ-assistent: Wictor Lund

# Vad skall vi lära oss

---

- Vad ett operativsystem är
- Vilka delar det är uppbyggt av
- Varför är det bra att känna till operativsystem?
- Behöver vi kunna operativsystem i framtiden?
- Undersöka operativsystem

# Uppläggning

---

- 1.9 Introduktion till kursen, vad är ett operativsystem? [pdf](#) [pdf](#)
- 3.9 OBS INGEN FÖRELÄSNING, föreläsare på resa
- 4.9 Programmeringsspråk för OS: C och assembler, Arkitektur, Modeller pdf
- 8.9 Processer / trådar pdf
- 10.9 Skedulering, deadlocks pdf
- 11.9 IPC `<-A href="forelasning_5.pdf">`pdf
- 15.9 Systemanrop och avbrott pdf
- 17.9 Minneshantering I pdf
- 18.9 Minneshantering II pdf
- 22.9 Minneshantering III pdf
  
- 24.9 Input / output I pdf
- 25.9 Input / output II pdf
- 1.10 INGEN FÖRELÄSNING: Föreläsare på resa/möte
- 3.10 INGEN FÖRELÄSNING: Föreläsare på resa/möte
- 4.10 INGEN FÖRELÄSNING: Föreläsare på resa/möte
- 6.10 Filsystem I / II pdf
- 8.10 Nätverk pdf
- 9.10 Multiprocessorsystem, distribuerade system pdf
- 13.10 Säkerhet pdf
- 15.10 Case Study I: Linux / Mac OS X pdf
- 16.10 Case Study II: Windows XP/Vista pdf NT kernel Internals lecture notes

# När behöver man kunna operativsystem?

---

- Skriva drivrutiner (Inbyggda system)
- Få handtelefon att fungera bättre
- Reda ut varför web-servern fungerar så långsamt
- Förstå massiv datalagring på ett företag



# Linux som studieobjekt

---

- Implementerar ett typiskt UNIX operativsystem (Linus Torvalds har i stort utgått från Tanenbaums läror och hans Minix operativsystem)
- Källkoden publikt tillgängligt, man kan utan problem undersöka och se hur implementerat
- Källkoden normalt installerat i katalogen
  - `/usr/src/linux-2.6/`
  - `/usr/src/kernel/<kernel-version>`där 2.6/kernel-version står för versionsnumret på operativsystemets kärna (kernel)

# Vad är ett operativsystem?

---

- Generellt sett:
  - Fungerar som "virtuell maskin" (VM, Virtual Machine) för den egentliga hårdvaran
    - Hårdvaran accesserbar via generella funktionsanrop som inte är maskinspecifika
    - Samma användarprogram kan köras på olika hårdvaror
  - Fungerar som resursallokator
    - Allokerar hårdvaruresurser mellan processer/job som körs i systemet
      - Tidsplanet: CPU åt en process i taget
      - Rymden: Centralminnet fördelas mellan processer

# Var används operativsystem?

---

- 1960-talet: "Operativsystemet är mjukvaran som kontrollerar hårdvaran", ett program åt gången
- Idag: Datorer exekverar många applikationer samtidigt: Operativsystemet skall erbjuda en säker och "rättvis" grund för applikationsprogram
- Allt fler apparater behöver operativsystem: MP3-spelare, PDA, mobiltelefoner, inbyggda system såsom komponenter i nätverk, skrivare, digitalkameror, TV-apparater, bilar
- Vilket är världens vanligaste operativsystem?
  - Nokia OS? Symbian? Win7? Android?

# Marknadsandelar, PC

Total Market Share



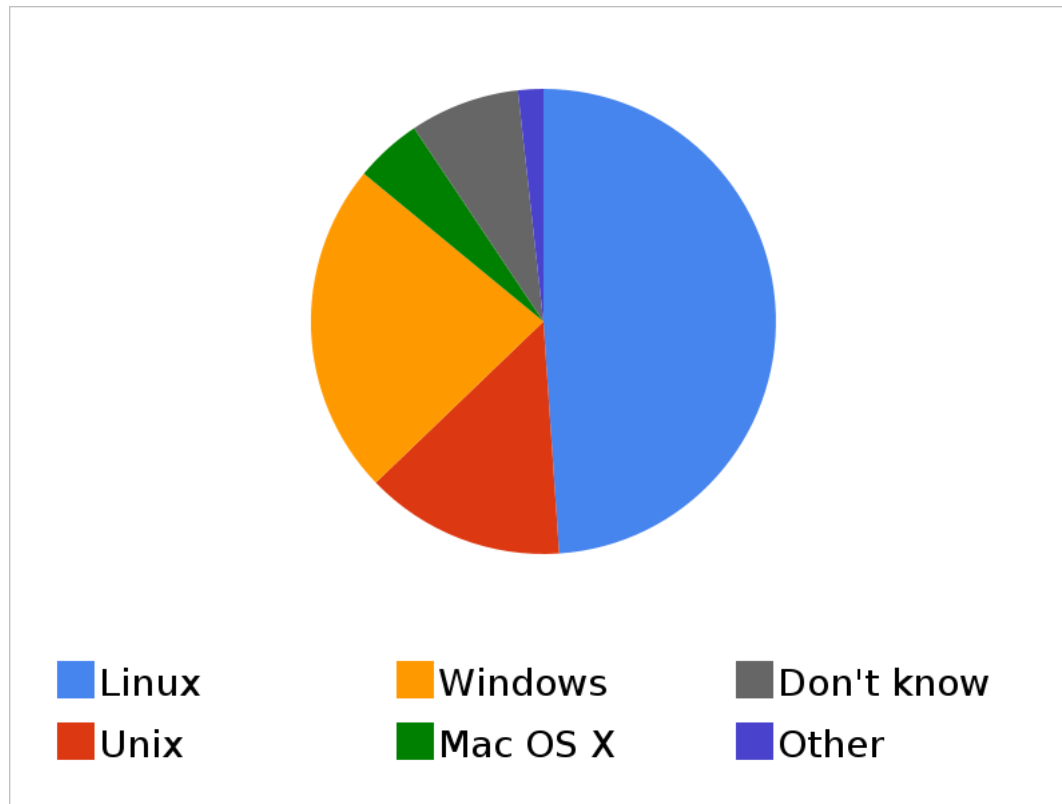
Total Market Share



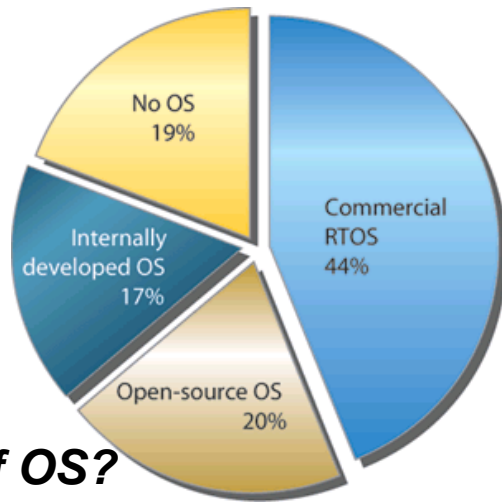
Källa: <http://marketshare.hitslink.com>

# Marknadsandelar, server

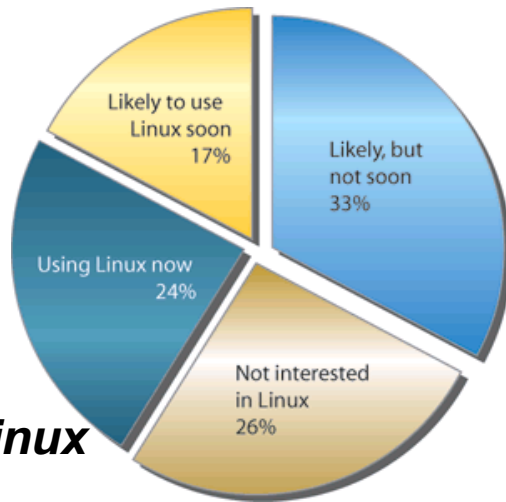
---



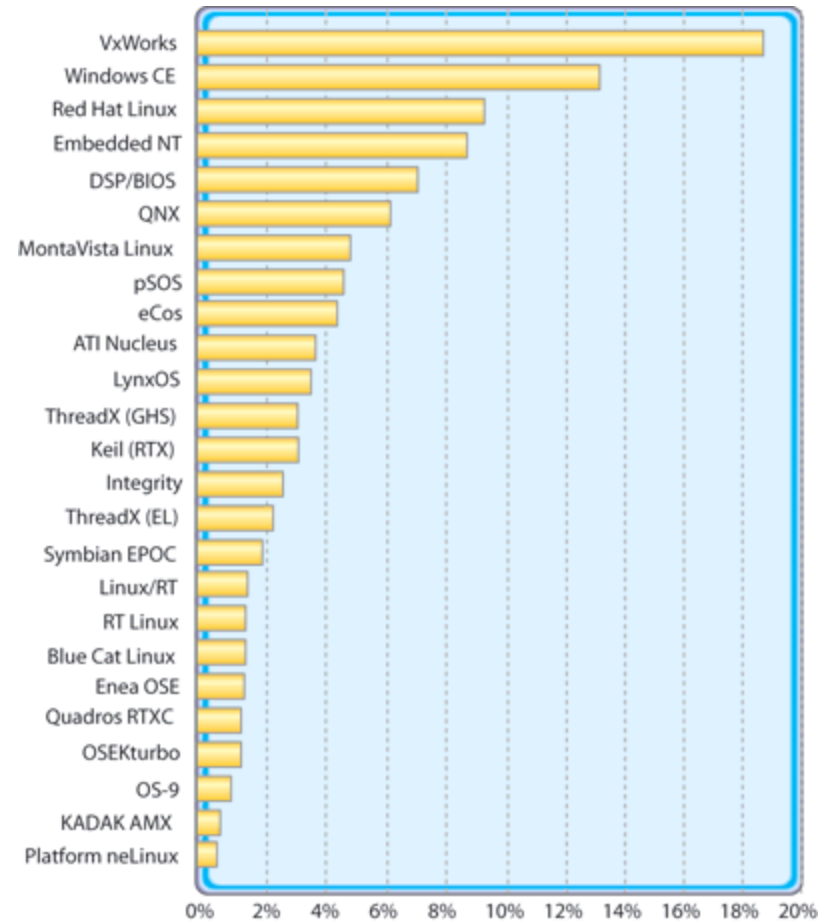
# Marknadsandelar - inbyggda



**What type of OS?**



**Interest in Linux**



**Current commercial OS**

# Krav på operativsystem

---

- Responstid och genomströmning
  - I medeltal
  - I värsta fall
- Robusthet
- "Rättvisa"
  - Alla processer skall få resurser
    - Minne, hårddiskenutrymme, nätverksbandbredd
- Säkerhet
  - Data skall finnas tillgänglig då den behövs
  - Data skall inte finnas tillgänglig för obehöriga
- Dynamik
  - Skall klara varierande situationer
    - T.ex. "hot-swapping" av hårddisken

# Omgivningar för operativsystem

---

- Bordsdatorer
  - Kontorsapplikationer
- Inbyggda system
  - Ofta en specifik uppgift
  - Begränsade (och statiska) hårdvaruresurser
  - Tillgänglighet!!
- Mission-critical systems
  - Processtekniska styrsystem: Garanterad funktionalitet
- Kommersiella tillämpningar
  - E-business: Snabb responstid
  - Intranet: Datatillgängligt
  - Bankverksamhet: Datasäkerhet

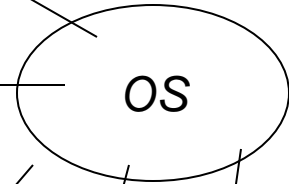


# Omgivning, sett ur annat perspektiv

---

- Mainframes
  - "Gammaldags" stordatorer, t.ex. inom bankväsendet, flygbokningar
  - Mycket I/O
- Serveromgivningar
  - Erbjuder resurser åt andra datorsystem
  - Datormoln
- Realtidssystem
  - Strikta responstider
- Inbyggda system
- Smart-cards

# Operativsystem



1950

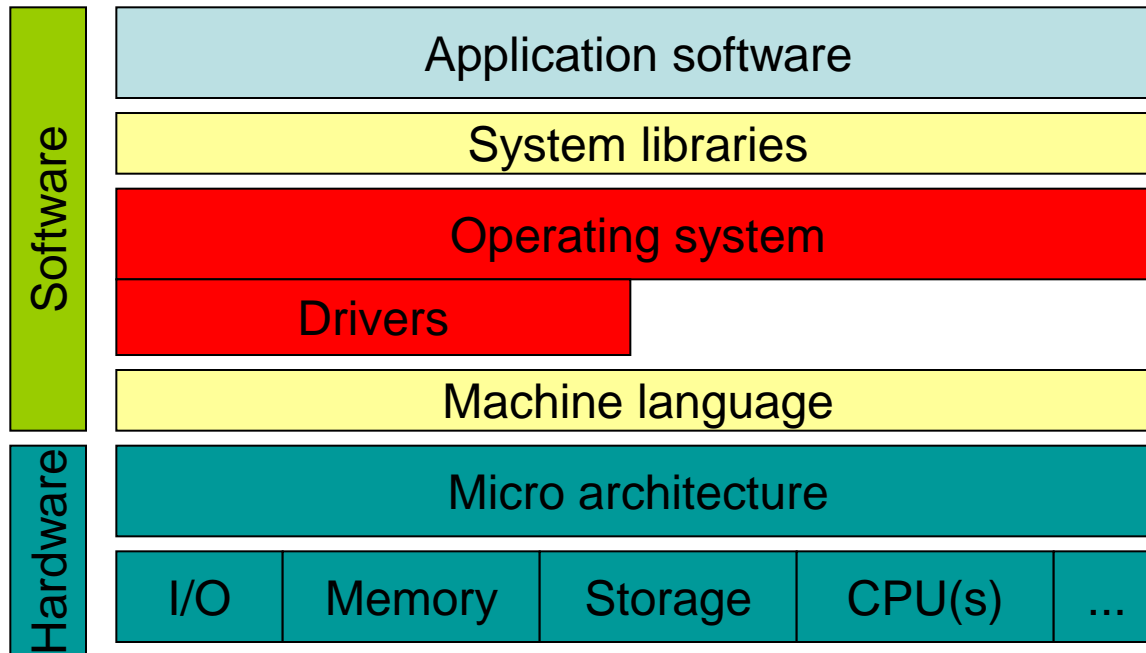
1980

1990

2000-

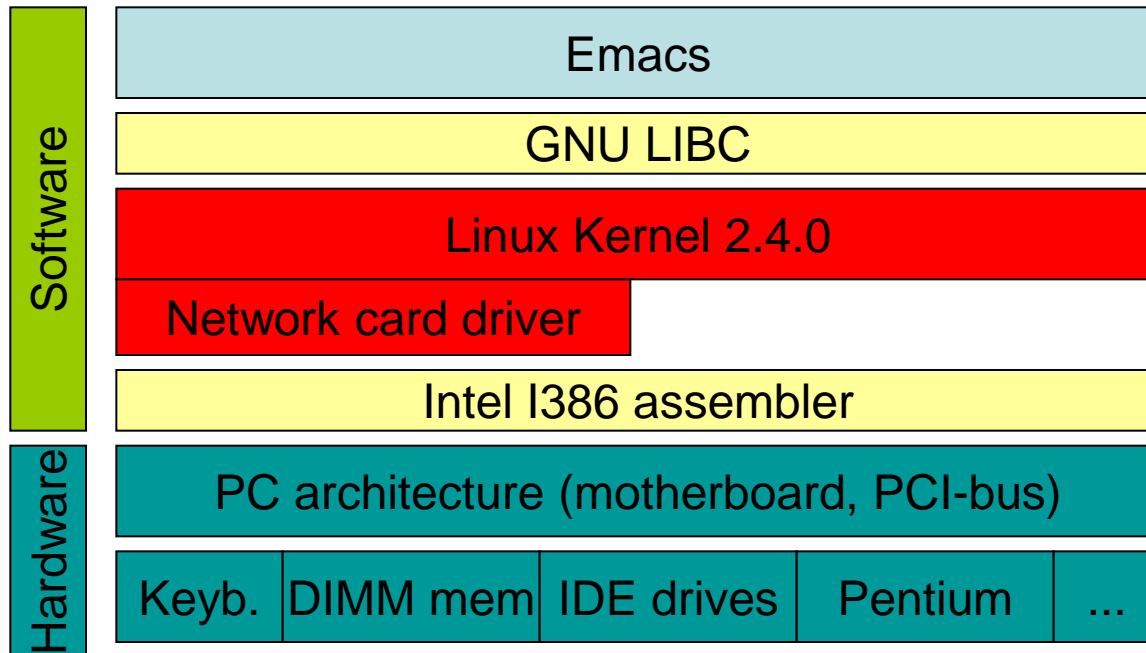
# Ett datorsystem

---



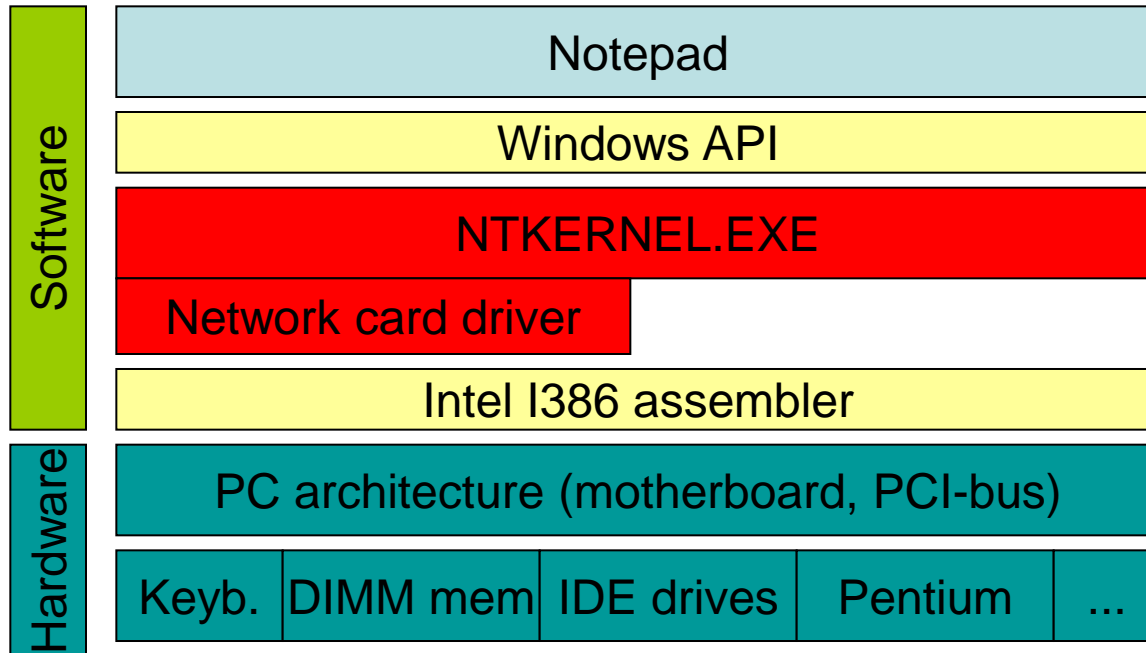
# Unix (Linux)

---



# Windows

---



# Exempel på systemanrop

---

- Då helloworld.c körs:

```
main() {  
    printf("Hello World\n");  
}
```

- 25 systemanrop (kernel-funktioner)

- Då HelloWorld.java körs

```
public class HelloWorld {  
    public static void main(String args[])  
    {  
        System.out.println("Hello  
World!");  
    }  
}
```

- 2248 systemanrop !!

# Operativsystemets grunddelar

---

- Processer / trådar
- Minneshantering
- Drivrutiner för hårdvara, Input / Output
- Filer / filsystem
- Nätverkshantering
- API (Application Program Interface), systembibliotek