



max planck institut  
informatik

## **Moderne Rechner**

## **Hardware and Software**

**Kurt Mehlhorn**

**Kosta Panagioutou**

**Max-Planck-Institut für Informatik**

# Inhalt

- Hardware = die Maschine, z.B. dieses Notebook lenovo X220 Tablet
- Software = Programme, die auf der Maschine laufen, z.B. Windows, Word, Powerpoint, Firefox, Thunderbird, Skype.
- In der Software liegt die Würze
- Maschinen sind rasend schnell



# AEG Computer Room, 1913



# Erste Computer



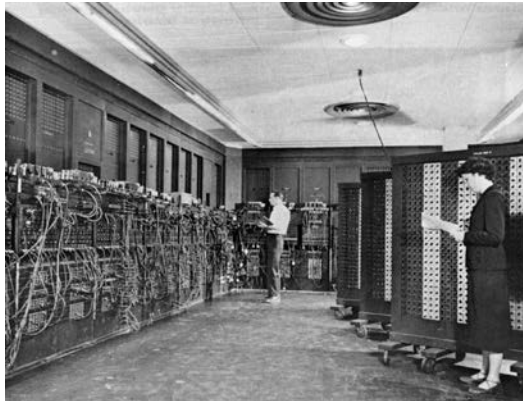
Zuse Z1 (1937)



Z3 (1941)



Z4 (1945)

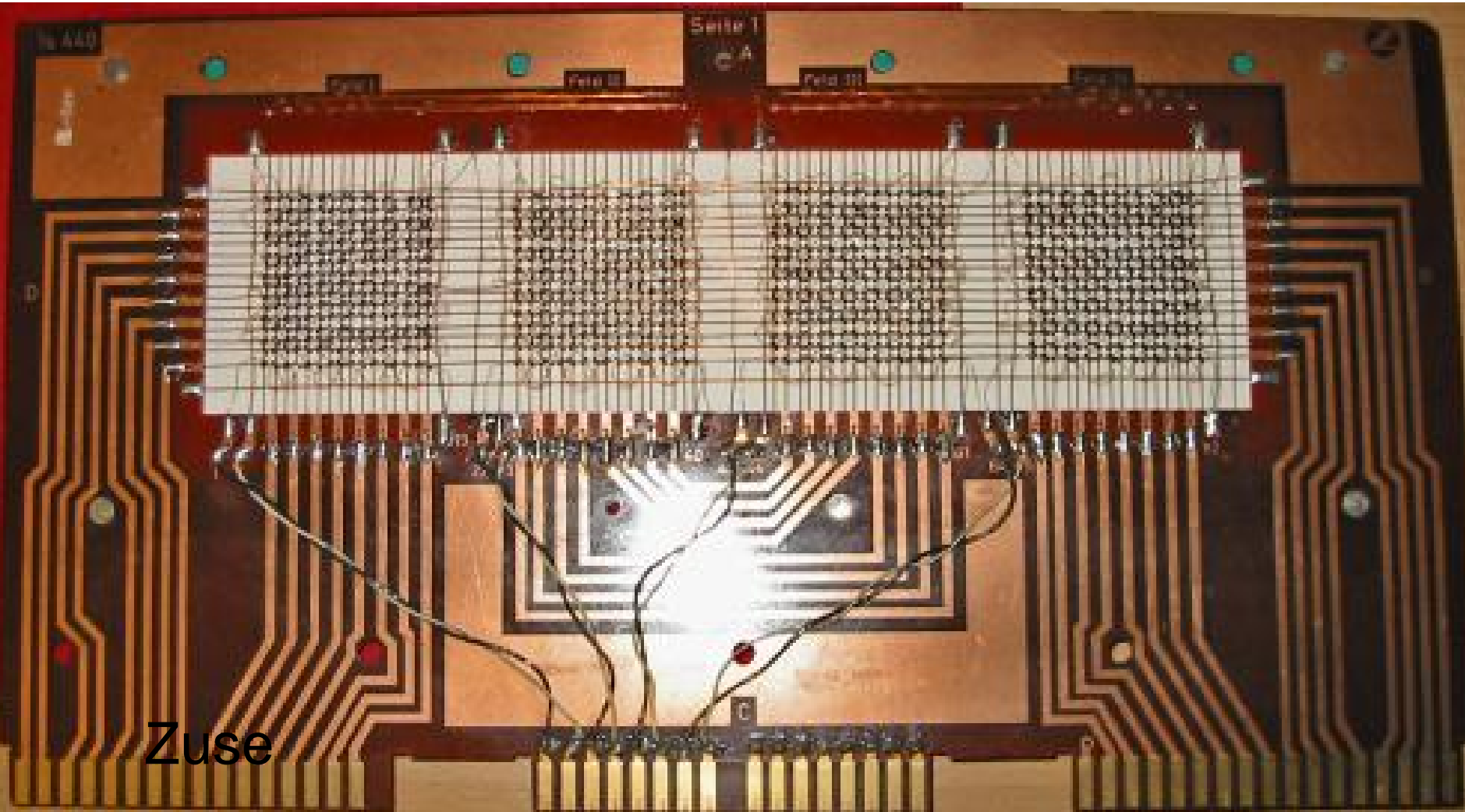


ENIAC (1946)

Z3, Z4 und ENIAC sind programmierbar (Programm extern) und Turingmächtig

Z3 und Z4 arbeiten mit Relais, ENIAC arbeitet mit Röhren

# Speicher



Zuse

Original Speicherchip v. Konrad Zuse, Erfinder des Computers, handgefertigt, ca. 25 x 14 cm groß

# Arithmetische Einheit





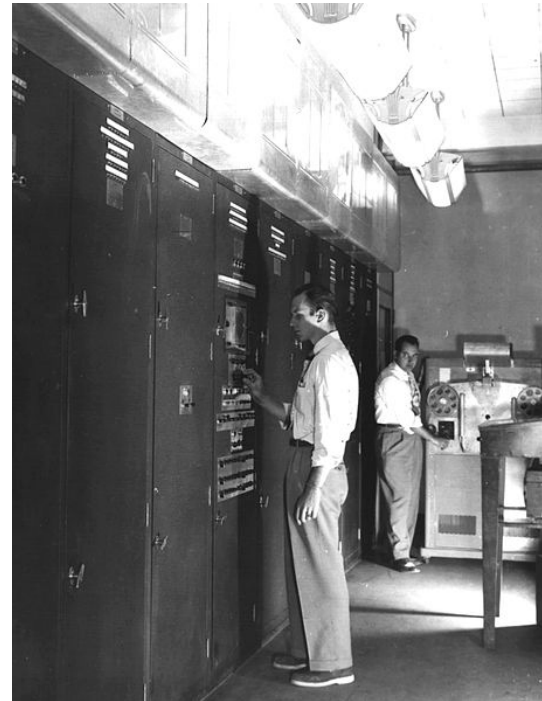
max planck institut  
informatik

# EDVAC (Electronic .... Computer)

First Draft of a Report on the EDVAC  
by John von Neumann,  
June 30, 1945

EDVAC, fertiggestellt in 1951

- Stored program
- Speicher, 5.5 kilobytes
- multiplication time 2.9 milliseconds
- 6,000 vacuum tubes
- consumed 56 kW of power
- 45.5 m<sup>2</sup> of floor space and weighed 17,300 lb (7,850 kg)
- operating personnel was thirty people for each eight-hour shift
- Kosten 500,000 Dollar (entspricht etwa 6 Millionen in 2010)

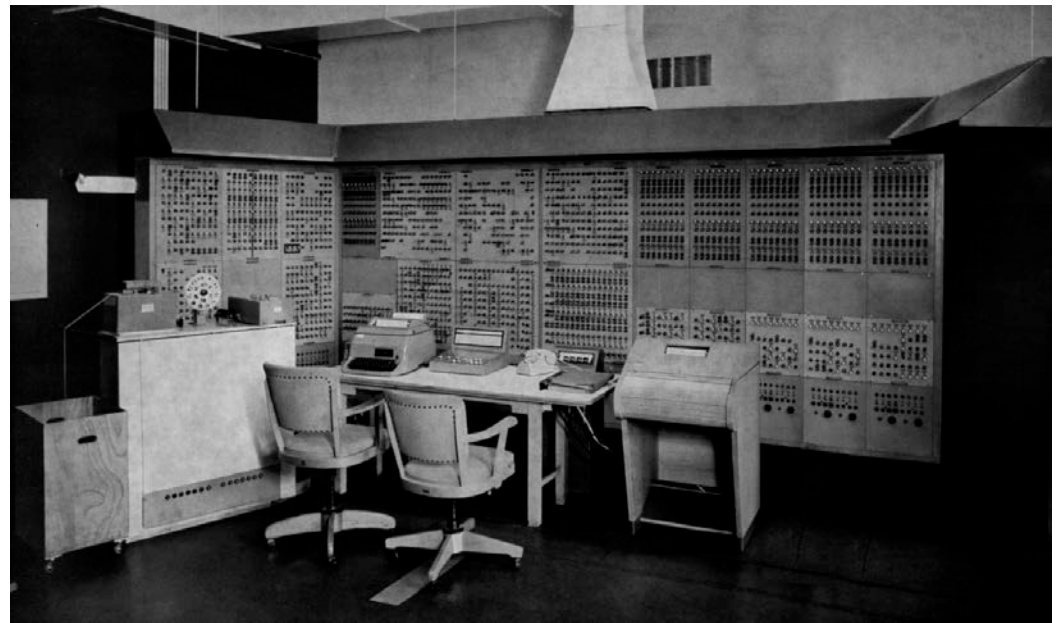


## Das Vorbild für alle modernen Rechner



# Mein erster Rechner

- Programmierbare Elektronische Rechenanlage München (PERM), röhrenbasiert
- Betriebnahme 7. Mai 1956, Piloty/Sauer
- erster ALGOL-Compiler
- KM lernte auf der PERM programmieren

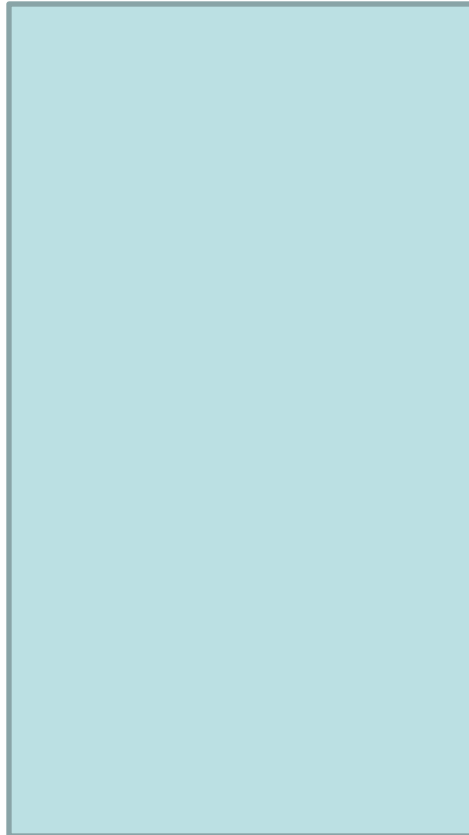


# PERM

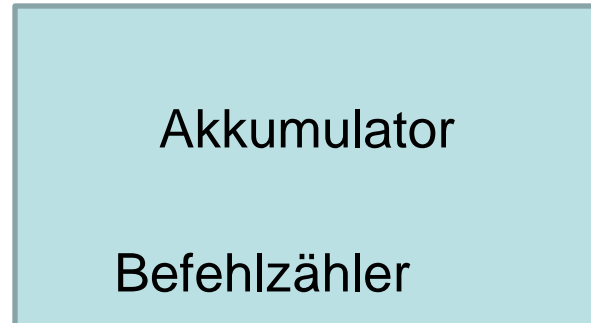


# Von Neumann Rechner

Speicher



CPU



- Speicher enthält Daten und Programm
- Befehlszyklus
  1. Führe Befehl mit Nummer BZ aus
  2. Erhöhe BZ um eins
  3. Gehe nach 1.

# Typische Befehle

- LOADZERO  $ACC \leftarrow 0$
- LOAD  $n$   $ACC \leftarrow M[n]$
- STORE  $n$   $M[n] \leftarrow ACC$
- ADD  $n$   $ACC \leftarrow ACC + M[n]$
- DECR  $ACC \leftarrow ACC - 1$
- JUMPPPOS  $n$  if  $ACC > 0$ ,  $BZ \leftarrow n$
- STOP

# In $M[1]$ steht eine Zahl $n$ , bilde $1 + \dots + n$

1. LOADZERO
2. STORE 2
3. LOAD 2
4. ADD 1
5. STORE 2
6. LOAD 1
7. DECR
8. STORE 1
9. JUMPPPOS 3
10. STOP

$M[1] = i$  und  $M[2] = i+1 + \dots + n$

$M[1] = i$  und  $M[2] = i + \dots + n$

$M[1] = i - 1$  und  $M[2] = i + \dots + n$

$M[1] = 0$  und  $M[2] = 1 + \dots + n$

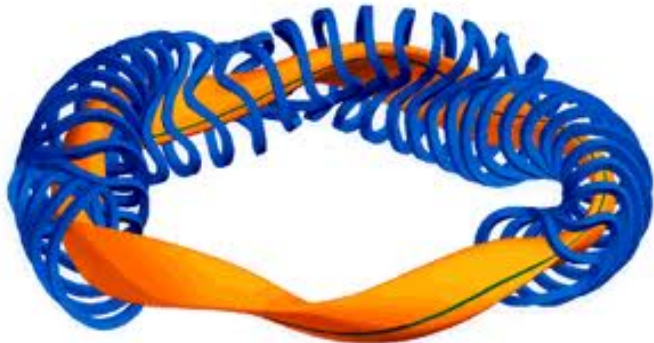
# Kenngrößen und Neuerungen

- Hauptspeicher:  $10^9$  Worte a 64 Bit
- Befehlszyklus:  $10^9$  Befehle pro Sekunde
- Eine Million mal leistungsfähiger (Geschwindigkeit, Speicher, Größe), eine Million mal billiger als 1950
- Neuerungen seit 1950
  - Interrupts (Unterbrechungen)
  - Speicherhierarchie: Cache, Main, Disk
  - **Bildschirme und Graphik**
  - **Netze**
  - Preis und Leistung
  - **Software, Nutzerfreundlichkeit**



# Supercomputer

- 72 Schränke,
- 73000 PowerPCs mit je 2 Gbyte RAM
- Simulation: Physik, Klima, Chemie, Strömung
- 13 Mio Euro



# Entstehung des Universums

## $10^{12}$ Partikel





# Bildschirme und Graphik

- Dieser Schirm: 1366 x 768 Pixels
- Für die CPU: für jedes der Pixel kann man Farbe und Helligkeit einstellen
- Graphikkarte zeigt die Werte auf dem Bildschirm an.

# Software

- Systemprogramme
  - Programmiersprachen und Compiler
  - Betriebssysteme: Viele Programme gleichzeitig
  - Internet
- Anwendungsprogramme
  - Büroprogramme
  - Email
  - Skype
  - Browser



# Programmiersprachen und Compiler

## C, C++, Java, Python,

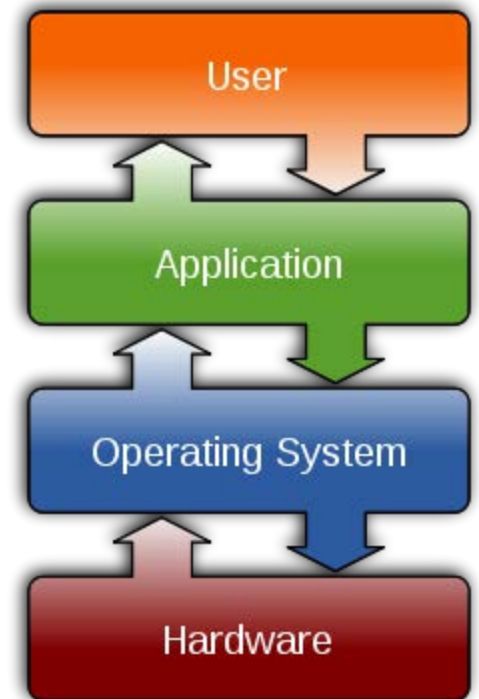
1. LOADZERO
2. STORE 2
3. LOAD 2
4. ADD 1
5. STORE 2
6. LOAD 1
7. DECR
8. STORE 1
9. JUMPPPOS 3
10. STOP

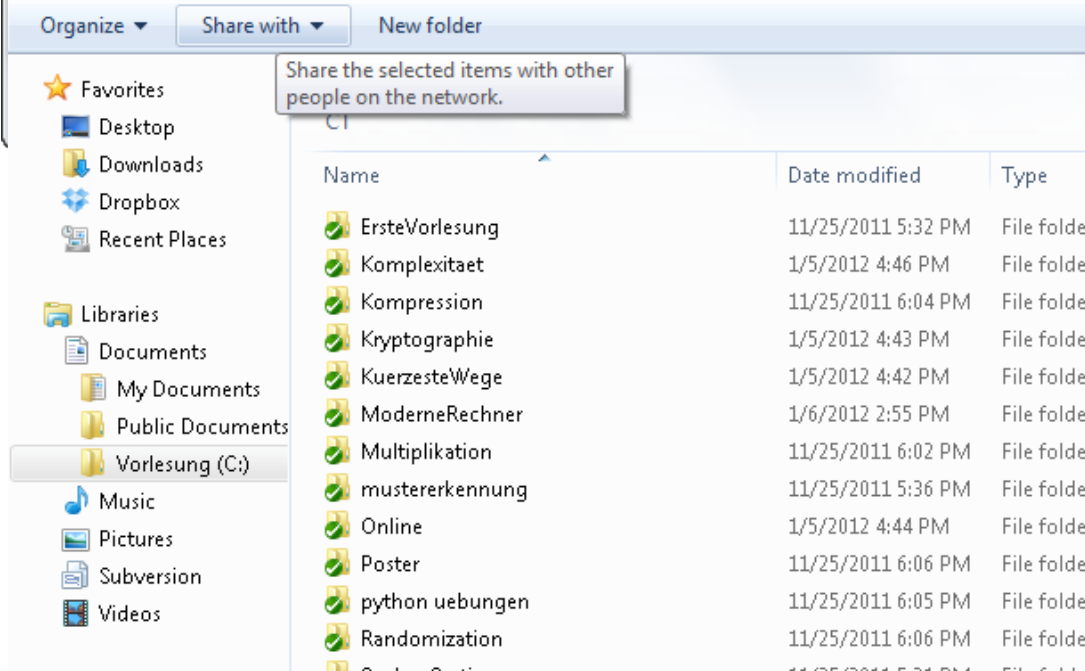
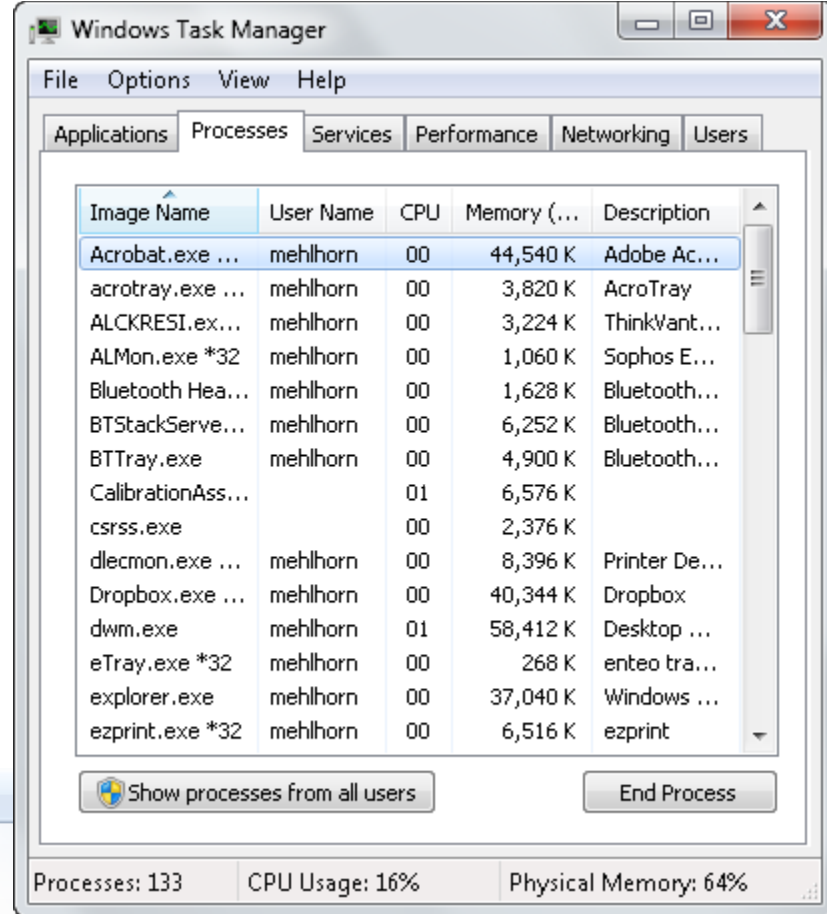
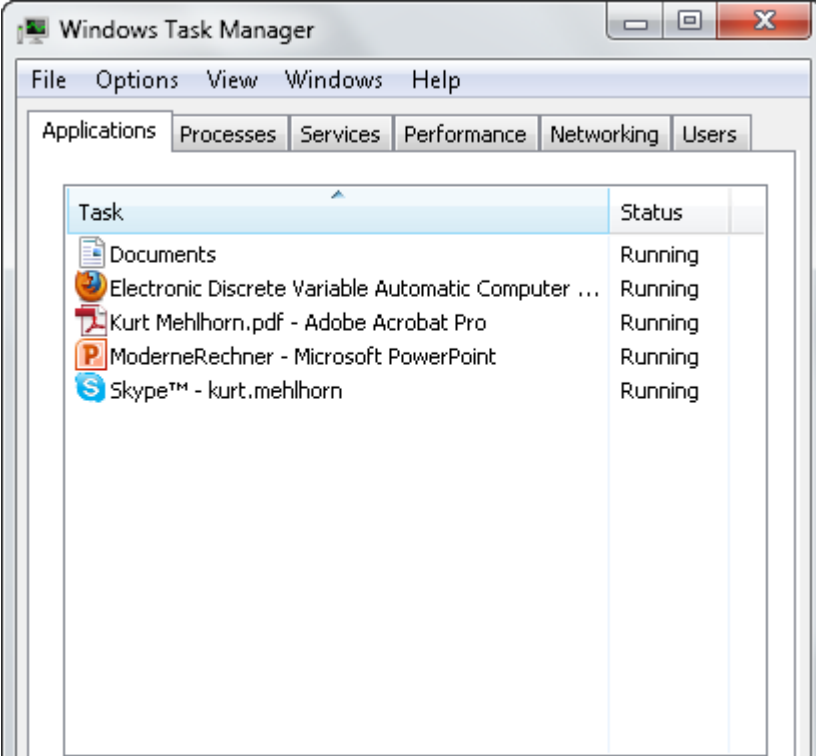
```
int Sum(int n)
{ int sum = 0;
  int i;
  for (i = 1; i <= n; i++)
    sum = sum + i;
  return sum;
}
```

# Betriebssystem (Operating System)

## Windows, Linux, Android, Mac-OS

- Hausmeister des Rechners
- Verwaltet Prozesse und Speicher
- Filesystem
- Interrupts, Unterbrechungen
- Gerätetreiber
- Ein- und Ausgabe
- Netzwerk
- User Interface

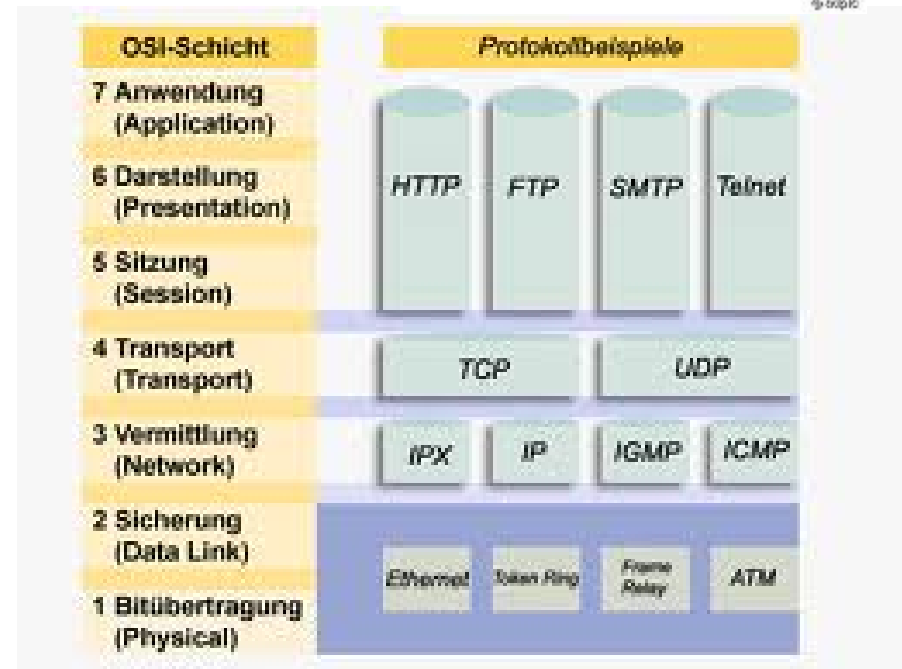




# Das Internet



- Internetdienste:  
Email, WWW, ...
- Anwendungsschicht:  
http, https
- Transportschicht,  
TCP
- Netzschicht, IP
- Physikalisches Netz



Protokoll = Regeln des korrekten Verhaltens

# Das Internet

<http://en.wikipedia.org/wiki/Wulff>

- Internetdienste: Email, WWW, ...
- Anwendungsschicht (Kommunikation)
- Transportschicht (Übertragungsprotokoll)  
zuverlässige Übertragung von Daten
- Netzschicht (Internetprotokoll): verschickt Pakete fester Länge von Punkt zu Punkt, best effort, keine Flusskontrolle
- Physikalisches Netz verschickt Bits über eine Leitung (auch Funk)



# Schicht 1: physikalische Schicht

- Daten werden über eine Leitung übertragen (auch Funk)
- Ähnlich:
  - Post leert Briefkasten und bringt sie zur Hauptpost
  - Post transportiert Briefe von einem Postamt zu einem anderen Postamt.



# Schicht 2: Internetprotokoll

- Rechner haben Namen, 137.253.17.34  
ähnlich zu Telefonnummern
- Geographisch gruppiert
- DNS (Telefonbuch) übersetzt von lesbaren  
Namen zu Nummern
- IP stellt Pakete fester Länge zu; Wege  
werden dynamisch bestimmt
- wie Briefe bei der Post

# Schicht 3: Übertragungsprotokoll

- Schickt Dateien zuverlässig von Punkt zu Punkt
- Bricht Dateien in Pakete auf
- Überprüft Zustellung
- Setzt Pakete wieder zur Datei zusammen (richtige Reihenfolge)
  
- Analogie: Sekretariate

# Anwendungsschicht

- Einfache grundlegende Anwendungen
- http = Hypertext Transfer Protokoll
- <http://en.wikipedia.org/wiki/Merkel>
- Findet Nummer von wikipedia bei DNS
- Schickt Nachricht GET wiki/Merkel an en.wikipedia.org
- En.wikipedia.org schickt Datei wiki/Merkel zurück

# Hohe Anwendung, z.B. Browser

```
<H1>Homepage</H1> <TABLE> <TBODY>
<TR vAlign=top align=left> <TD> <a
href="fotos/Kurt1980.jpg"> </a> </TD> <TD> <H2>Kurt
Mehlhorn</H2> <P><A title="Max-Planck-
Insitut für Informatik" href="http://www.mpi-
inf.mpg.de/">Max-Planck-Institut für
Informatik</A><BR> <A title="Department 1:
Algorithms and Complexity"
href="http://www.mpi-
inf.mpg.de/departments/d1">Department 1:
Algorithms and Complexity</A><BR> <p> <a
href="http://frweb.cs.uni-
sb.de/index.php?theme=1&lang=en">Compu
ter Science Department, Saarland
University</a> <p> <a
href="http://www.mmci.uni-saarland.de/"
">Cluster of Excellence: Multimodal
Computing and Interaction</a> </p><p> <a
href="http://www.impecs.org">Indo Max
Planck Center for Computer Science</a>
```

Kurt Mehlhorn

http://www.mpi-



mpi

max planck  
informatik

## Homepage



### Kurt Mehlhorn

Max-Planck-Institut für  
Informatik  
Department 1: Algorithms  
and Complexity

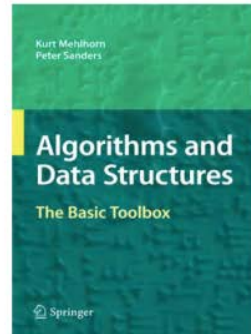
Computer Science  
Department, Saarland  
University

Cluster of Excellence:  
Multimodal Computing  
and Interaction

Indo Max Planck Center  
for Computer Science

**Phone:** +49 681 9325  
100 (within Campus: 92  
100)

**Fax:** +49 681 9325 199  
(within Campus: 92 199)



max planck institut  
informatik

Office hours: Tue. 3-4 pm

[Information for prospective students and postdocs](#)

Teaching (Winter Term 2011/12)

# Schreibprogramme (Word, Latex)

- Automatischer Randausgleich,  
Proportionalschrift statt Schreibmaschine,  
Ligaturen
- Ich gehe nach Hause und komme übermorgen wieder ins Büro um dann weiterzuarbeiten. Also dann bis übermorgen.
- Ich gehe nach Hause und komme übermorgen wieder ins Büro um dann weiterzuarbeiten. Also dann bis übermorgen.
- ff statt  $\text{f f}$

# Email

- Elektronische Post ist eine Nachbildung der Funktionalität klassischer Post
- Eine wesentliche Neuheit sind Spamfilter; darüber mehr in der Vorlesung über maschinelles Lernen

# Empfehlungssysteme

- Kunden, die dieses Buch gekauft haben, haben auch folgende Bücher gekauft
- Wenn ihnen diese fünf Filme gefallen haben, dann sollten ihnen auch dieser Film gefallen
- Thunderbird glaubt, dass diese email Spam ist.

# Kunden, die dieses Buch ...

Für jedes Buch  $i$  speichere Paare

( $j$ , Anzahl der gemeinsamen Verkäufe von  $i$  und  $j$ )

- Zwei Kopien der Liste: einmal sortiert nach Buchnamen, einmal sortiert nach Anzahl der Verkäufe



# Zusammenfassung

- Rechner sind extrem schnell:  $10^9$  Befehle pro Sekunde
- Software ist in Schichten aufgebaut
- Schichten bieten nach oben Dienste an und nutzen Dienste der nächsten Schicht
- Viele wichtige Entwicklungen von Einzelpersonen oder kleinen Teams