

# Organisatorisches

- Übung Blatt 8 am 22.12. ganz normal
- Übung Blatt 8 auch am 8.1. vor der Vorlesung (Antonios).
- Vorlesung wieder am 8.1. (Antonios)
- Übung am 12.1. fällt aus
- Übung Blatt 9 am 15. und 19.1 (Dr. Cosmina Croitoru)
- Vorlesung am 15.1. (Dr. Michael Sargaloff)
- Vorlesung am 22.1. (Prof. Dr. Kurt Mehlhorn)
- Ich werde diese Woche noch eine Musterklausur online setzen.



max planck institut  
informatik

# Ideen und Konzepte der Informatik

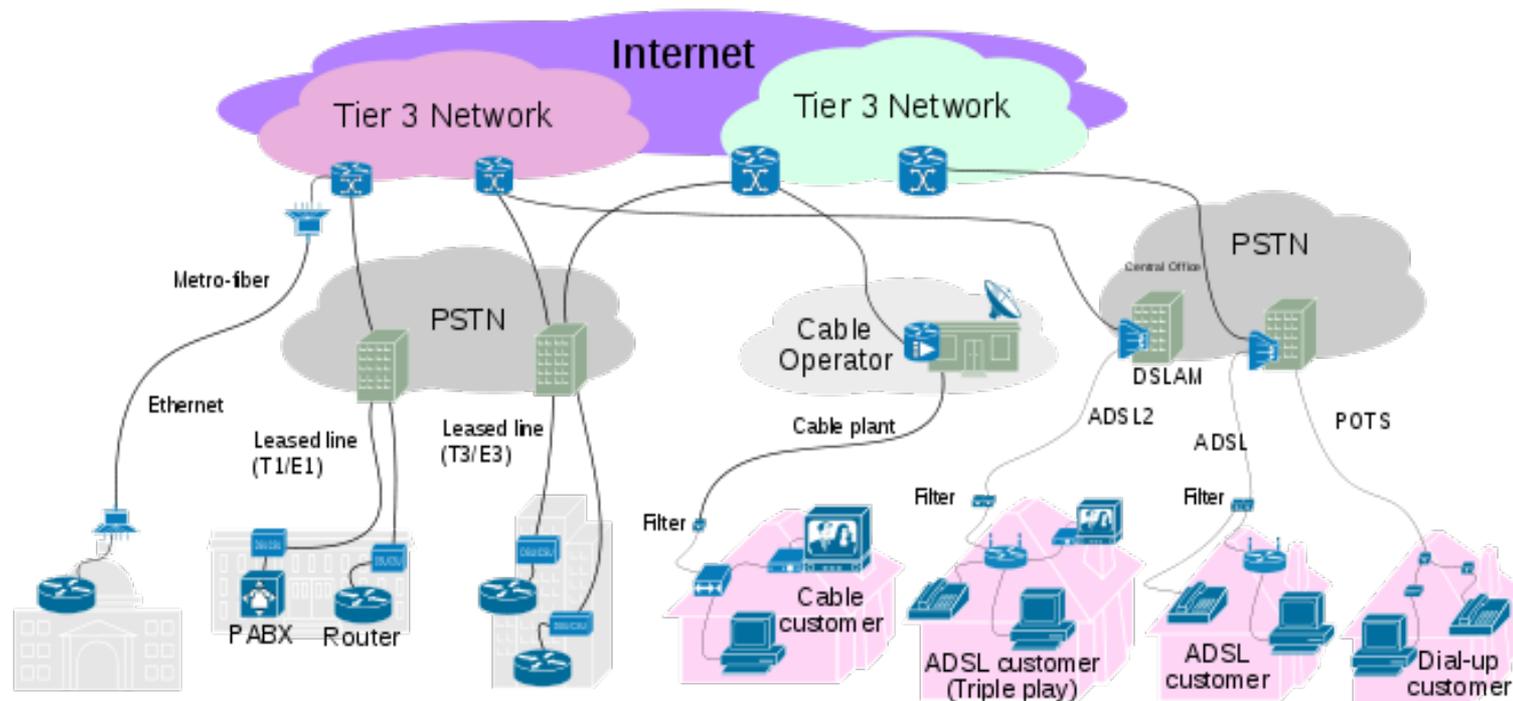
## Das Internet

**André Nusser**

(Foliensatz von Kurt Mehlhorn)

# Was passiert alles,

- wenn ich eine Webseite aufrufe?
- wenn ich eine E-Mail abschicke?



# Überblick

- Datenübertragung
  - zwischen zwei Rechnern
  - zwischen Rechnern in einem Netzwerk
  - zwischen Netzen im Internet
- Aufbau von Webseiten
- Darstellung im Webbrowser
- E-Mail

# Datenübertragung

- Bits werden als Spannung am Kabel übertragen, z. B.  
 $+5V = 1$ ,  $-5V = 0$
- ... Oder per WLAN
- ... Oder per Satellit
- ... Oder per Brieftaube
- Unterschiede müssen für den Benutzer unsichtbar sein!

# Konstruieren in Schichten

- Eine Schicht (Layer) bietet Dienste an höhere Schichten an und nutzt die Dienste der darunterliegenden Schicht zur Realisierung. Realisierung ist nach oben hin verborgen.
- Unterste Schicht setzt auf der physikalischen Realität auf.
- Klempner nutzt Rohre, Zangen, Bohrmaschine und bietet Installationsdienst für Häuser. Architekt nutzt Installationsdienst und bietet Bäder. Normen erleichtern die Zusammenarbeit

# Schichten

- Link Layer
  - Abstrahiert von der Technik im lokalen Netz, von der Physik zum Bit
- Internet Layer
  - Verbindet das lokale Netz mit dem Netzanbieter, Transport ohne Garantien, vom Bit zu Paketzustellung
- Transport Layer
  - Fehlertolerante Datenübertragung
- Data Layer
  - Kommunikationsprotokoll zwischen Browser und Server, Dienste für den Endnutzer

# Ethernet, ein populäres Netzwerk

- Kabelgebunden
- $+5V = 1$ ,  $-5V = 0$
- 100 – 1000 Millionen Bits pro Sekunde

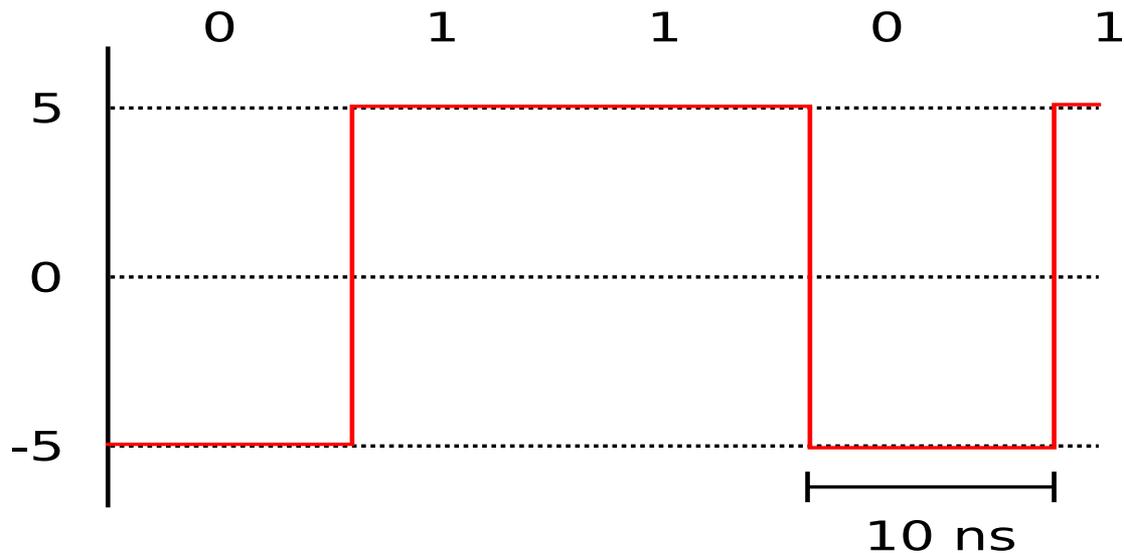


Abbildung ist stark idealisiert

# Probleme

- Uhren:
  - Wann messe ich die Spannung?
  - Welche Uhrenqualität braucht man?
  - 1 000 000 Einsen =  $10^{-2}$  Sekunden 5V, nicht  $10^{-2}$  Sekunden + 10 ns
- Störungen
  - Sollte das eine 1 sein, oder hat jemand den Föhn angemacht?

# Selbstsynchronisierung

## billige Uhren tun's auch

- Uhren mit Nanosekundenpräzision sind teuer.
- Lösung: Nie zu lange 1 oder 0 senden, z. B.

### Manchester-Kodierung:

- Kodiere 0 als 01 und 1 als 10
- Also 0001101 als 01010110100110
- In der kodierten Folge nie mehr als 2 gleiche Symbole hintereinander; Unterscheidung von 1 und 2 Takten reicht; selbstsynchronisierend

# Störungen

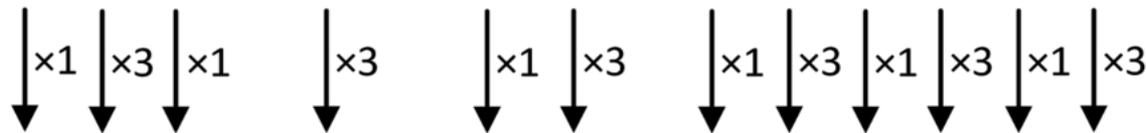
- Übertragungsfehler passieren ständig
  - 1 Fehler pro 10 Millionen Bits = 10 Fehler/s
- Meistens: Viele Bits hintereinander falsch
- Bits werden in Pakete zusammengefasst
- Jedes Paket bekommt eine Prüfsumme; siehe nächste Folie
- Bei Fehlern im Paket: Neuübertragung

# Prüfsummen

- Einfachste Prüfsumme = Quersumme
- besser (Zahlendreher): gewichtete QS

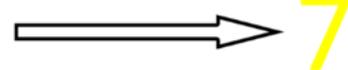
Beispiel: Prüfziffer bei der ISBN-13

9 7 8 - 3 - 1 2 - 7 3 2 3 2 0 - ?



$$9 + 21 + 8 + 9 + 1 + 6 + 7 + 9 + 2 + 9 + 2 + 0 = 83$$

Abstand zum  
nächsthöheren  
Vielfachen von 10



# Fehlerkorrigierende Codes (Reed-Solomon)

- Ich will  $k$  Zahlen senden, z. B.  $k = 128$ ; ich sende Zahlen statt Bits, weil das die Mathematik einfacher macht.
- Ich sende  $k + 2d$  Zahlen.
- Bis zu  $d$  Zahlen dürfen bei der Übertragung korrumpiert werden. Trotzdem kann der Empfänger die  $k$  Zahlen rekonstruieren.
- Ich zeige das Prinzip für  $k = 2$  und  $d = 2$ . Es gibt auch noch Folien für  $k = 3$  und  $d = 2$  zum Selbststudium.

# Mathematischer Hintergrund ( $k = 2$ )

- Eine Gerade ist durch zwei Punkte bestimmt.
- Durch zwei beliebige Punkte geht eine Gerade.
- Stimmen zwei Geraden an zwei Punkten überein, so sind sie gleich.
- Zwei verschiedene Gerade schneiden sich höchstens einmal.

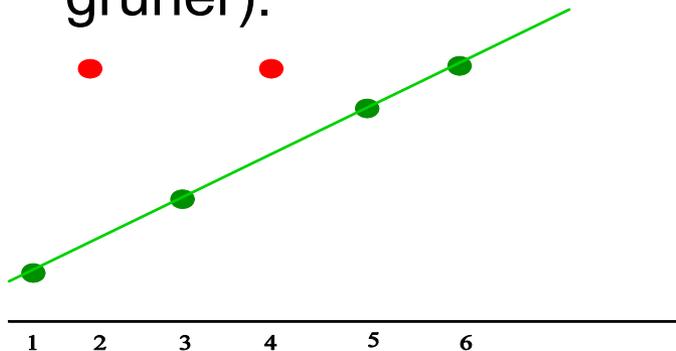
# Fehlerkorrigierende Codes (Reed-Solomon)

- Ich will 1 2 senden.
- Bestimme die eindeutige Gerade  $p$  mit  $p(1) = 1$ ,  $p(2) = 2$ .
- $p(x) = x$ .
- Sende 1 2  $p(3) = 3$ ,  $p(4) = 4$ ,  $p(5) = 5$ ,  $p(6) = 6$ .
- Bei der Übertragung passieren 2 Fehler. Der Empfänger erhält

1 6 3 6 5 6

# Fehlerkorrigierende Codes (Reed-Solomon)

- Der Empfänger erhält 1 6 3 6 5 6. Für jedes Paar von Werten bestimmt er die Gerade. Es gibt  $15 = 5 \cdot 6/2$  Paare.
- $p(1) = 1, p(3) = 3 \rightarrow$  richtige Gerade
- $p(1) = 1, p(4) = 6 \rightarrow$  falsche Gerade
- Auf der richtigen Gerade liegen 4 (grüne) Punkte. Auf einer falschen Gerade liegen höchstens 3 Punkte (zwei rote und ein grüner).



Also wird die richtige Gerade öfter gefunden als jede falsche.

**Mehrheitsentscheid**

# Ein Geheimnis teilen

- Möchten Bob und Alice ein Geheimnis geben, so dass es einer allein nicht rekonstruieren kann.
- Sei  $g$  das Geheimnis. Wähle eine zufällige Zahl  $a$  und gib Bob die Zahl  $g - a$  und Alice die Zahl  $g + a$ .
- Zusammen können sie  $g$  bestimmen, da  $(g - a + g + a)/2 = g$ .
- Einer allein weiß gar nichts:  $g + a$  ist eine zufällige Zahl.

# Mathematischer Hintergrund ( $k = 3$ )

- Ein Polynom vom Grad  $< 3$  ist durch seine Werte an drei Stellen eindeutig bestimmt.
- Stimmen zwei Polynome vom Grad  $< 3$  an drei Stellen überein, so sind sie gleich.
- Für drei Stellen darf man die Werte beliebig vorgeben: Interpolationspolynom.
- Zwei verschiedene Polynome vom Grad  $< 3$  schneiden sich höchstens zweimal.

# Mathematischer Hintergrund (k = 3)

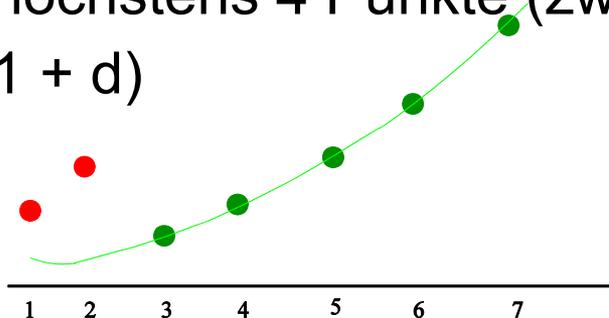
- Ein Polynom vom Grad  $< 3$  ist durch seine Werte an drei Stellen eindeutig bestimmt.
- $p(x) = a_2x^2 + a_1x + a_0$ , Polynom vom Grad  $< 3$ ;  
 $a_2, a_1, a_0$  sind die Koeffizienten.
- $p(5) = 25a_2 + 5a_1 + a_0$
- Falls  $p(0)=2, p(2)=16, p(-1)=4$ , dann  $a_2 = 3, a_1 = 1, a_0 = 2$ .

# Fehlerkorrigierende Codes (Reed-Solomon)

- Ich will 1 1 3 senden.
- Bestimme das eindeutige Polynom vom Grad  $< 3$  mit  $p(1) = 1, p(2) = 1, p(3) = 3$ .
- $p(x) = x^2 - 3x + 3$
- Sende 1 1 3  $p(4) = 7, p(5) = 13, p(6) = 21, p(7) = 31$ .
- Bei der Übertragung passieren 2 Fehler. Der Empfänger erhält  
4 7 3 7 13 21 31.

# Fehlerkorrigierende Codes (Reed-Solomon)

- Der Empfänger erhält 4 7 3 7 13 19 31. Für jedes Tripel von Werten interpoliert er. Es gibt 35 Tripel.
  - $p(3) = 3, p(5) = 13, p(7) = 31 \rightarrow$  richtiges Polynom
  - $p(1) = 4, p(5) = 13, p(7) = 31 \rightarrow$  falsches Polynom
  - Auf dem richtigen Polynom liegen mindestens 5 Punkte (mindestens  $k + d$ ). Auf einem falschen Polynom liegen höchstens 4 Punkte (zwei rote und zwei grüne, allgemein  $k - 1 + d$ )
- Daher wird das richtige Polynom öfter gefunden als jedes falsche.



**Mehrheitsentscheid.**

# Ein Geheimnis teilen

- Möchte  $n$  Personen ein Geheimnis geben, so dass es je  $k$  rekonstruieren können, aber  $k - 1$  es nicht können.
- Sei  $g$  das Geheimnis. Wähle zufällige Zahlen  $a_1$  bis  $a_{k-1}$  und bestimme das eindeutige Polynom  $p$  vom Grad  $< k$  mit  $p(0) = g$  und  $p(i) = a_i$  für  $1 \leq i \leq k - 1$ .
- Gib der  $i$ -ten Person das Paar  $(i, p(i))$ ,  $1 \leq i \leq n$ .
- Anwendung:  $g$  ist ein Schlüssel. Je  $k$  Teilnehmer können schließen, aber keine  $k - 1$  können es.

# MAC (media access control) Adressen

- Im Ethernet hört jeder alles auf der Leitung.
- Konfliktauflösung
- Jedes Gerät hat eine eindeutige MAC Adresse (von Geburt an).
- Datenpakete haben einen Adresspräfix.  
Prozessor holt sich die für ihn bestimmten Nachrichten von der Leitung.



# Internet Protocol (IP)

- Bietet Paket-Kommunikation *zwischen* Netzwerken
- Egal ob die Technik gleich ist oder nicht (Ethernet vs. WLAN).
- Best Effort, keine Garantien:
  - Pakete gehen verloren
  - Pakete kommen doppelt an
  - Reihenfolge kann sich ändern

# IP Adressen

- Wie Telefonnummern für Computer
- 32 Bits für die Adresse (inzwischen 128 Bits)
  - Vier Zahlen zwischen 0 und 255
  - Zum Beispiel *139.19.14.56 = MPI-INF*
  - Regionales Clustering
  - Hat man nicht von Geburt an (wie bei der MAC-Adresse), sondern bekommt man zugewiesen
- Ungefähr 4 Milliarden mögliche Adressen

# IP Routing

- Jeder Router (Verteiler) hat eine Tabelle

Ziel	Link	Distanz
192.168.*.*	1	15
192.169.*.*	2	5
192.170.*.*	1	12

- Ist Ziel in meinem Netz? Direkt an MAC.
- Sonst in der Tabelle nachschlagen und auf entsprechendem Ausgabelink weiterleiten.

# Routing Information Protocol

- Das Netz ändert sich ständig, z. B. Reparaturen oder neue Hardware.
- Router berechnen kontinuierlich kürzeste Pfade im Netz (kurz = wenige Hops).
- Alle 30 Sekunden: Tabelle an alle Nachbarn weiterreichen.
- Update: Wenn mein Nachbar einen deutlich besseren Weg zu einem Ziel kennt, schicke ich die entsprechenden Pakete in Zukunft an ihn.

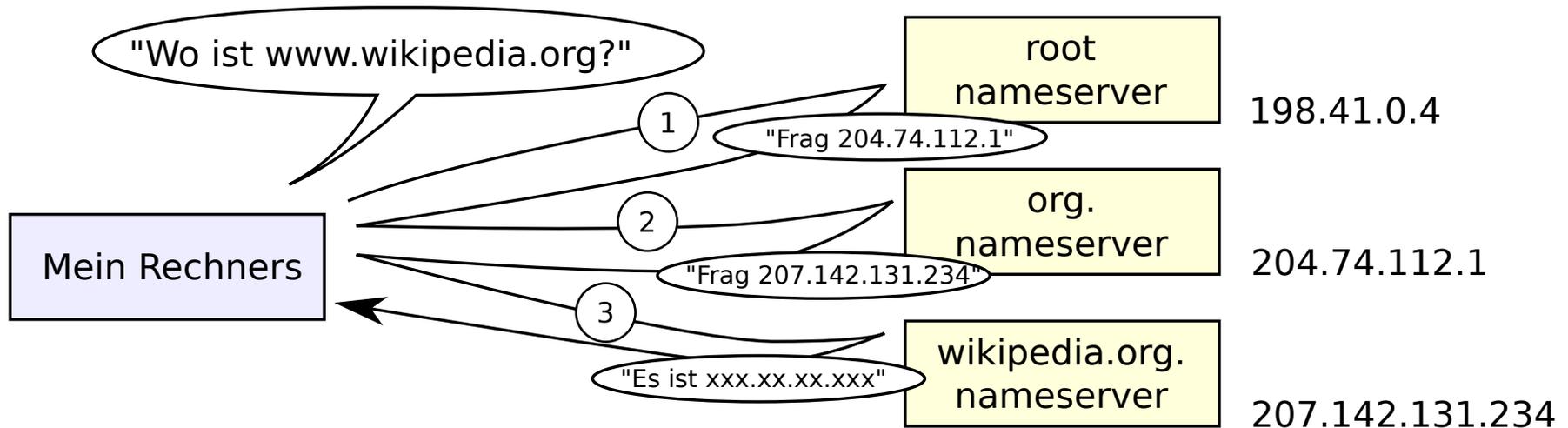
# Transmission Control Protocol (TCP)

- Zuverlässige Datenübertragung zwischen Rechnern
  - Pakete nummerieren → Reihenfolge
  - Pakete mit Rückschein
  - Bleiben Bestätigungen aus → Neu senden

# DNS

- Telefonbuch für IP Adressen
  - Übersetzt *www.google.de* in 173.194.35.151
- „Nameserver“ speichern Tabellen
  - Tabelle enthält entweder Paar (Name, IP).
  - Oder Verweis auf Nameserver (mit .de gehst du besser zur Telekom).
  - Lokales Telefonbuch versus Auskunft.
- Jeder Computer hat eine Liste mit Nameservern.

# Nachschnlagen von Wikipedia.org



Man geht zuerst zum Root-Nameserver. Der verweist einen weiter.

# Zwischenstand

---

- Ethernet und WLAN, um im lokalen Netzwerk zu reden.
- IP, um zwischen Netzwerken Pakete zu schicken.
- TCP, um zuverlässig über IP zu reden.
- DNS, um IP Adressen nachzuschlagen.

# E-Mail

- Post an *andre.nusser@mpi-inf.mpg.de* schicken.
- Mailprogramm fragt Nameserver nach *mpi-inf.mpg.de* und schickt die E-Mail an *mpi-inf.mpg.de*.
- *mpi-inf.mpg.de* speichert alle E-Mails an *andre.nusser* in dessen Postfach.
- Ich hole sie von dort ab.

# Hypertext Transfer Protocol, HTTP

- HTTP ist ein Protokoll zur Übertragung von Daten auf der Anwendungsschicht über ein Rechnernetz.
- Es wird hauptsächlich eingesetzt, um Webseiten (Hypertext-Dokumente) aus dem World Wide Web (WWW) in einen Webbrowser zu laden.
- Webseiten sind in HTML kodiert.

# Hypertext (HTML)

- „Sprache“, in der Webseiten beschrieben sind.
- Der Text legt die Struktur der Webseite fest (Überschriften, Gliederung in Abschnitte, Tabellen, ... ) aber nur die ungefähre Darstellung.
- Webseiten enthalten Text, Bilder, Verweise, klickbare Objekte, ...
- Browser berechnet Details der Darstellung, etwa Zeilenumbrüche, .....

# Ausschnitt aus Prof. Mehlhorns Webseite

## <A>Books and Book Chapters</A></H2>

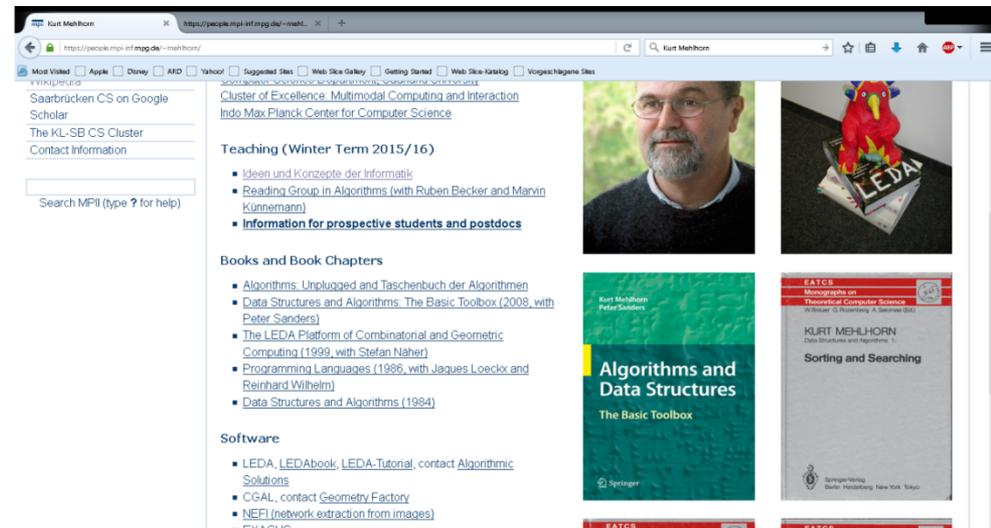
### <UL type=circle>

<li><a href="AlgorithmsUnplugged.html">Algorithms: Unplugged and Taschenbuch der Algorithmen </a></li>

<li><a href="Toolbox.html">Data Structures and Algorithms: The Basic Toolbox (2008, with Peter Sanders) </a></li>

<li><a href="LEDAbook.html">The LEDA Platform of Combinatorial and Geometric Computing (1999, with Stefan Näher) </a></li>

</ul>



# Dynamische Elemente

- Mausbewegungen, Klicks etc. werden vom Betriebssystem verwaltet
- Browser wird über „Events“ benachrichtigt
- Darstellung kann sich dynamisch ändern
  - Seite muss (effizient!) neu gezeichnet werden
- Klicken löst Aktionen aus
  - Zum Beispiel werden Videos abgespielt

# HTTPS versus HTTP

- http: unverschlüsselte Übertragung. Problematisch bei offenen WLANs
- S = secure
- Bietet
  - Authentifizierung der Partner
  - Verschlüsselte Kommunikation
- Empfehlung: HTTPS Everywhere benutzen

# Zusammenfassung

