

Netzwerk



Stable Diffusion Art “Networks”

Inhaltsverzeichnis

1. Internet und Adressierung	3
1.1. Historisches	3
1.2. Wiederholung Bits, Bytes und Hex	3
1.3. IP-Adressen	5
1.4. Adressvergabe	6
2. Topologie von Rechnernetzen	6
3. Protokolle	6
4. Das Schichtenmodell	6
5. Nebenläufige Prozesse	6
6. Das klassische Erzeuger - Verbraucher - Problem	6
7. Verklemmungen	6



For now I 1:1 copied the structure of the pptx I have here. I will also mostly adapt the content, if you have Bedenken right away, here we go

1. Internet und Adressierung

1.1. Historisches

Das **Internet** entstand primär aus dem 1969 entwickelten ARPANET (Advanced Research Project Agency Network) Projekt an dem vier US-Amerikanische Universitäten beteiligt waren, deren Ziel die weitläufige Vernetzung von Rechnern zur gemeinsamen Nutzung von Ressourcen war. Die Idee größere Netzwerke aufzubauen stammte dabei schon aus den Anfängen der 1960er Jahre und viele Gruppen forschten und arbeiteten daran (z.B. auch das US Militär). Dieses Projekt war auch eines der ersten, die das **TCP/IP** Protokoll implementiert haben ([RFC 9293](#) - der aktuelle Standard).

In der Folge wurden vor allem weitere Universitäten in Europa und den USA in Netzwerken miteinander verknüpft bis die Technologie so weit ausgereift war, dass es zu einer regelrechten Explosion kam, die dazu führte, dass die meisten technischen Geräte heute miteinander **physikalisch** verbunden sind.



Please feel free to add information you deem important at any point, but please be aware that we do only have so much time to discuss things

Das **World Wide Web** ist in den 80er Jahren am CERN (Conseil européen pour la recherche nucléaire) entwickelt worden und *nicht* synonym zum Internet zu verstehen. Spricht man vom Internet, dann ist primär die physikalische Verbindung, d.h. die Hardware der Netzwerke gemeint. Das WWW ist jedoch ein Dienst, d.h. eine Software. Die Grundlage bilden dabei aufeinander verweisende Hypertext-Dokumente, die auf verschiedensten Rechnern abgespeichert sind. Das WWW benutzt also das Internet als Medium, ist jedoch nur einer von vielen Diensten, die das tun. Weitere Beispiele für Dienste sind **E-Mails**, **FTP** (file transfer protocol) oder das **XMPP** (extensible messaging and presence protocol - für Chat-Anwendungen).

1.2. Wiederholung Bits, Bytes und Hex

In der Informatik werden andere **Stellenwertsysteme** als in der schulüblichen Mathematik verwendet. Während dort das Zehnersystem Einzug gehalten hat ist für Computer das Zweiersystem geeignet (Strom an, Strom aus!).

Das bedeutet konkret: Statt in Zehnerpotenzen zu denken und daraus die Zahlen “zusammenzubauen” wie in der fünften Klasse (z.B. $12985 = 10000 + 2000 + 900 + 80 + 5$) werden **Binärzahlen** aus **Zweierpotenzen** zusammengebaut.

Das erscheint uns auf den ersten Blick merkwürdig - das liegt letztendlich aber nur an der Gewöhnung an das Zehnersystem.

Zweierpotenzen für die Umrechnung									
512	256	128	64	32	16	8	4	2	1

Beispiele

- Binär: 0111 1010 lässt sich dezimal schreiben als:

$$0 \cdot 128 + 1 \cdot 64 + 1 \cdot 32 + 1 \cdot 16 + 1 \cdot 8 + 0 \cdot 4 + 1 \cdot 2 + 0 \cdot 1 = 122$$

- Dezimal: 195 lässt sich umwandeln zu:

$$195 = 1 \cdot 128 + 1 \cdot 64 + 0 \cdot 32 + 0 \cdot 16 + 0 \cdot 8 + 0 \cdot 4 + 1 \cdot 2 + 1 \cdot 1$$

und entspricht somit in Binärdarstellung als 1100 0011

Beschränkt man sich auf Zahlen, die durch 8 Bit (also 1 Byte!) dargestellt werden, dann ist der Zahlenraum also auf die Zahlen von 0 bis 255 beschränkt, da 255 binär 1111 1111 entspricht.

Aufgabe

Wandeln Sie jeweils in Binär bzw. Dezimaldarstellung um.

- 253
- 1110 0111
- 187
- 1011
- 11 0101 1011

Hexadezimale Darstellung von Zahlen

Große 0-1 Folgen sind für Menschen schwer zu lesen. Zum Test - welcher Dezimalzahl entspricht die folgende Binärzahl?

01010001101011110000110011101000

Es ist natürlich 1370426600! Um solche 0 – 1-Folgen lesbarer zu machen wird häufig die **hexadezimale** Darstellung von Zahlen verwendet, diese verwendet nicht das Zweiersystem, sondern das “Sechzehnersystem”. Da wir mit vier Stelle einer Binärzahl gerade 16 Zahlen darstellen können, teilen wir also die obige Zahl in Viererblöcke ein:

0101 0001 1010 1111 0000 1100 1110 1000

Jeder dieser Viererblöcke entspricht jetzt einer “Ziffer” im Sechzehnersystem. Da es zu Uneindeutigkeiten kommen würde, wenn man “10” als Ziffer verwendet (Dieses Problem hatte man beim Zweiersystem natürlich nicht) müssen andere Zeichen als “Ziffern” verwendet werden. Man hat sich hier auf die Zeichen a bis f geeinigt, es gilt also:

$(0000)_2 = (0)_{16}$	$(0001)_2 = (1)_{16}$	$(0010)_2 = (2)_{16}$	$(0011)_2 = (3)_{16}$
$(0100)_2 = (4)_{16}$	$(0101)_2 = (5)_{16}$	$(0110)_2 = (6)_{16}$	$(0111)_2 = (7)_{16}$
$(1000)_2 = (8)_{16}$	$(1001)_2 = (9)_{16}$	$(1010)_2 = (a)_{16}$	$(1011)_2 = (b)_{16}$
$(1100)_2 = (c)_{16}$	$(1101)_2 = (d)_{16}$	$(1110)_2 = (e)_{16}$	$(1111)_2 = (f)_{16}$

Die Klammern zeigen dabei an, wie die Zeichen innerhalb interpretiert werden sollen (das hätte man auch schon bei der Umwandlung vom Binär- ins Dezimalsystem machen können, dort ist die Mehrdeutigkeit aber nicht so dramatisch.)

Die obige Binärzahl entspricht also der folgenden Hex-Zahl:

5 1 a f 0 c e 8

Häufig gruppiert man diese noch zu Zweierpaketen, dann entspricht eine dieser Hex-Gruppen genau einem Byte ($2 \cdot 4$ Bit!)

51 af 0c e8

Nochmal veranschaulicht:

51 af 0c e8



Look at the rainbow. LOOK AT IT!

0101 0001 1010 1111 0000 1100 1110 1000

Aufgabe

Wandeln Sie jeweils von Binär- in Hexadezimaldarstellung oder umgekehrt um:

- 0110 0111
- *ff af*
- 1010 1100 0001 1001
- 0e c1 10

1.3. IP-Adressen

IP(v4) ist die erste Version des Internet Protocols, das weltweit verbreitet und eingesetzt wurde. Eine solche IP-Adresse besteht aus einer 32-stelligen Binärzahl, die jeweils einer Komponente (einem Gerät oder Netzwerk) im Internet zugeordnet ist und so eine Identifizierung möglich macht.

Zur besseren Lesbarkeit wird die Adresse üblicherweise in vier 8-stellige Böcke aufgeteilt und (für den Menschen) dann in Dezimalschreibweise abgebildet werden - jeweils durch einen Punkt getrennt.

Beispiel

01101100001000010000111100110111

01101100 00100001 00001111 00110111

108.33.15.55

Kurze Denkaufgabe!

Wie viele solcher IP-Adressen können gebildet werden?

Es sind natürlich 2^{32} , also etwa 4,3 Milliarden Adressen.

Aufgabe

Wandeln Sie jeweils von Binär- in Dezimaldarstellung oder umgekehrt um:

- 88.215.213.26
- 0101 1000 0110 0011 0011 1111 1001 0011

IP(v6) Da die Anzahl der möglichen IPv4-Adressen doch recht klein ist im Vergleich zu der Anzahl an Geräten und Netzwerken wurde ein neuer Standard entwickelt. Die Adressen in diesem Standard sind 128 Bit lang, somit gibt es 2^{128} unterschiedliche IPv6-Adressen.

Hier kommt auch die hexadezimale Darstellung ins Spiel, denn die Größe der Zahlen macht es sonst wieder unlesbar. Die Adresse wird also in 8 Blöcke zu je 16 Bit eingeteilt, die wiederum hexadezimal kodiert werden, a.so z.B.:

2600 : 1f18 : 001f : db00 : 807b : f1f4 : d01b : 30b1

Hinweis

Ob dieses Format wirklich lesbarer ist oder wirklich sein muss sei einmal dahingestellt..



As I understood Ipv6 does not play a huge role anyway does it? Why is that so? I want a justification for why this is very short here basically

1.4. Adressvergabe

IP-Adressblöcke werden an Unternehmen und Organisationen vergeben, die diese ihrerseits wieder verteilen können.

Vergebene Adressblöcke: [Findet man z.B. hier](#)

2. Topologie von Rechnernetzen

3. Protokolle

4. Das Schichtenmodell

5. Nebenläufige Prozesse

6. Das klassische Erzeuger - Verbraucher - Problem

7. Verklemmungen