

Aufgabe 4.1: JSON

a. Machen Sie sich mit dem [JSON](#) Format vertraut. Welche Vor- und welche Nachteile sehen Sie verglichen mit XML?

- Vorteile
 - einfache Syntax, Handhabung, Implementierung
 - einfach (maschinen)lesbare Textform
 - gültige JSON-Dokumente können in JavaScript mittels eval() interpretiert werden.
 - Unabhängigkeit von einer bestimmten Programmiersprache (Parser in fast allen bekannten Programmiersprachen)
 - Gute Integration in viele Web-Programmiersprachen
 - Geringer Overhead
- Nachteile
 - geringere Typmächtigkeit, z.B. unterstützt JSON nicht alle von JavaScript unterstützen Datentypen wie NaN, Infinity und -Infinity
 - keine Erweiterbarkeit
 - keine Attributreihenfolge
 - [unterstützt (wie XML) keinen Binärdatentyp]

b. Formulieren Sie die Inhalte [des] folgenden XML-Dokumentes in JSON:

```
{
  "course": {

    "name": [
      {
        "lang": "English",
        "description": "Computer Networks I"
      },
      {
        "lang": "German",
        "description": "Netzwerke I"
      }
    ],

    "typ": "L",

    "member": [
      {
        "id": 1088,
        "firstname": "Michaela",
        "lastname": "Meier"
      },
      {
        "id": 1090,
        "firstname": "Josef",
        "lastname": "Kantner",
        "visiting": ""
      }
    ],
  },
}
```

```

    "location": {
      "room": "R0.058",
      "street": "Lothstr. 64",
      "city": "Munich"
    }
  }
}

```

Aufgabe 5.1 SMTP

5.1.a)

“Nutzen Sie dann diese Verbindung, um an Ihren eigenen E-Mail Account eine E-Mail zu senden, die in einem Standard-Mail-Client auf folgende Weise angezeigt wird:

- Als Absender: Sarah Meier <sarah@kindergarten.de>
- Als Empfänger: Walter Weihnachtsmann <weihnachtsmann.walter@heaven.org>
- Als Betreff: Wunschzettel - bitte beachten
- Sowie ein kurzer Nachrichtentext Ihrer Wahl”

Client-seitige Telnet-Eingaben sind mit “\$ ” markiert:

```

$ telnet mailrelay4.rz.fh-muenchen.de 25
Trying 129.187.244.104...
Connected to mailrelay4.rz.fh-muenchen.de.
Escape character is '^J'.
220 mailrelay4.rz.fh-muenchen.de ESMTP Postfix
$ HELO cs.hm.edu
250 mailrelay4.rz.fh-muenchen.de
$ MAIL FROM: <zell@hm.edu>
250 2.1.0 Ok
$ RCPT TO: <zell@hm.edu>
250 2.1.5 Ok
$ DATA
354 End data with <CR><LF>.<CR><LF>
$ From: Sarah Meier <sarah@kindergarten.de>,
$ To: Walter Weihnachtsmann <weihnachtsmann.walter@heaven.org>
$ Subject: Wunschzettel - bitte beachten
$ Date: Tue, 10 Nov 2015 17:42:20 +0100
$ User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:38.0) Gecko/20100101
$ Thunderbird/38.3.0
$ Content-Type: text/plain; charset="utf-8"
$ Content-Transfer-Encoding: 7bit
$ MIME-Version: 1.0
$
$ Hi W,
$
$ ich will heuer nichts.
$
$ Danke
$ M
$ .
250 2.0.0 Ok: queued as 28ACE385CAC93

```

\$ QUIT

221 2.0.0 Bye

Connection closed by foreign host.

5.1.b)

“Lassen Sie sich dann in Ihrem E-Mail Client alle Header der Mail anzeigen. Machen Sie einen Screenshot und markieren Sie,

- welche auf den von Ihnen per SMTP-Protokoll an den Server gesendeten Informationen beruhen (d.h. von Ihnen so eingegeben wurden oder auf den von Ihnen eingegebenen Informationen ganz oder teilweise basieren)
- welche von (SMTP-)Servern automatisch (ohne Ihr Zutun) eingefügt wurden.”

Received: from BADWLRZ-SW13MB6.ads.mwn.de (2001:4ca0:0:108::154) by
BADWLRZ-SW13MB3.ads.mwn.de (2001:4ca0:0:108::151) with Microsoft SMTP Server
(TLS) id 15.0.1130.7 via Mailbox Transport; Tue, 10 Nov 2015 18:28:12 +0100
Received: from postforw2.mail.lrz.de (2001:4ca0:0:116::a9c:63e) by
BADWLRZ-SW13MB6.ads.mwn.de (2001:4ca0:0:108::154) with Microsoft SMTP Server
(TLS) id 15.0.1130.7; Tue, 10 Nov 2015 18:28:11 +0100
Received: from mailout2.lrz.de (lxmhs66.srv.lrz.de [IPv6:2001:4ca0:0:116::a9c:6a6])
by postforw2.mail.lrz.de (Postfix) with ESMTP id 3nwGP75pyTzyZJ
for <zell@hm.edu>; Tue, 10 Nov 2015 18:28:11 +0100 (CET)
Received: from lxmhs66.srv.lrz.de (localhost [127.0.0.1])
by mailout2.lrz.de (Postfix) with ESMTP id 3nwGP75XY6zyVD
for <zell@hm.edu>; Tue, 10 Nov 2015 18:28:11 +0100 (CET)
X-Virus-Scanned: by amavisd-new at lrz.de in lxmhs66.srv.lrz.de
Received: from mailout2.lrz.de ([127.0.0.1])
by lxmhs66.srv.lrz.de (lxmhs66.srv.lrz.de [127.0.0.1]) (amavisd-new, port 10002)
with ESMTP id 0fon8fiun6wP for <zell@hm.edu>;
Tue, 10 Nov 2015 18:28:11 +0100 (CET)
Received: from mailrelay4.rz.fh-muenchen.de (mailrelay4.rz.fh-muenchen.de [129.187.244.104])
by mailout2.lrz.de (Postfix) with ESMTP id 3nwGP721pSzybP
for <zell@hm.edu>; Tue, 10 Nov 2015 18:28:11 +0100 (CET)
Received: from localhost (localhost [127.0.0.1])
by mailrelay4.rz.fh-muenchen.de (Postfix) with ESMTP id 32D54385CCC11
for <zell@hm.edu>; Tue, 10 Nov 2015 18:28:11 +0100 (CET)
X-Virus-Scanned: amavisd-new at hm.edu
Received: from mailrelay4.rz.fh-muenchen.de ([127.0.0.1])
by localhost (mailrelay4.rz.fh-muenchen.de [127.0.0.1]) (amavisd-new, port 10024)
with LMTP id q1ta6Pvnqjox for <zell@hm.edu>;
Tue, 10 Nov 2015 18:28:11 +0100 (CET)
Received: from cs.hm.edu (b150.hm.vpn.lrz.de [129.187.52.150])
by mailrelay4.rz.fh-muenchen.de (Postfix) with SMTP id 28ACE385CAC93
for <zell@hm.edu>; Tue, 10 Nov 2015 18:27:34 +0100 (CET)
From: Sarah Meier <sarah@kindergarten.de>
To: Walter Weihnachtsmann <weihnachtsmann.walter@heaven.org>
Subject: Wunschzettel - bitte beachten
Date: Tue, 10 Nov 2015 17:42:20 +0100
Content-Type: text/plain; charset="utf-8"
Content-Transfer-Encoding: 7bit
Message-ID: <20151110172811.32D54385CCC11@mailrelay4.rz.fh-muenchen.de>
Return-Path: martin.zell@hm.edu

X-MS-Exchange-Organization-Network-Message-Id: be714ba9-d509-436e-7ce9-08d2e9f44ee7
X-MS-Exchange-Organization-AVStamp-Mailbox: Sophos;-1716141566;0;PM
X-MS-Exchange-Organization-AuthSource: BADWLRZ-SW13MB6.ads.mwn.de
X-MS-Exchange-Organization-AuthAs: Anonymous
MIME-Version: 1.0

Hi W,

ich will heuer nichts.

Danke
M

5.1.c)

“Recherchieren Sie die Bedeutung der in Ihrem Screenshot aus b) sichtbaren Header-Einträge und geben Sie für jeden Header-Eintrag in einem kurzen Satz an, wofür er verwendet wird.”

Header-Einträge	Bedeutung
Received: from BADWLRZ-SW13MB6.ads.mwn.de (2001:4ca0:0:108::154) by BADWLRZ-SW13MB3.ads.mwn.de (2001:4ca0:0:108::151) with Microsoft SMTP Server (TLS) id 15.0.1130.7 via Mailbox Transport; Tue, 10 Nov 2015 18:28:12 +0100	Die „Received“-Zeilen zeigen den Weg, den die E-Mail vom Sender zum Empfänger genommen hat. Jeder Server, der die Mail weiterleitet, fügt seine Kennung und das Datum am Anfang der E-Mail hinzu.
X-Virus-Scanned: by amavisd-new at lrz.de in lxmhs66.srv.lrz.de	Zeigt an, dass ein Virus-Scan durchgeführt wurde.
From: Sarah Meier <sarah@kindergarten.de>	Zeigt den (manipulierten) Absender an. Kann vom “echten”, wie in diesem Fall von zell@hm.edu, abweichen.
To: Walter Weihnachtsmann <weihnachtsmann.walter@heaven.org>	Zeigt den (manipulierten) Empfänger an. Kann vom “echten”, wie in diesem Fall von zell@hm.edu, abweichen.
Subject: Wunschzettel - bitte beachten	Betreff der Nachricht
Date: Tue, 10 Nov 2015 17:42:20 +0100	Absendedatum
Content-Type: text/plain; charset="utf-8"	Art und Zeichensatz des Body-Text
Content-Transfer-Encoding: 7bit	Zeichenkodierung für den Übertragungsweg.
Message-ID: <20151110172811.32D54385CCC11@mailrelay4.rz.fh-muenchen.de>	Eindeutige Zeichenfolge, die diese E-Mail identifiziert.
Return-Path: martin.zell@hm.edu	Gibt die Adresse an, an die im Falle eines Fehlers der Mailserver dazu nutzt, die Benachrichtigung zuzusenden.
X-MS-Exchange-Organization-Network-Message-Id: be714ba9-d509-436e-7ce9-08d2e9f44ee7	Eindeutige Zeichenfolge für das Netzwerk und die Nachricht.

X-MS-Exchange-Organization-AVStamp-Mailbox: Sophos;-1716141566;0;PM	Organisations-X-Header, der den "Stempel" der Antivirensoftware (AV). In diesem Fall Sophos.
X-MS-Exchange-Organization-AuthSource: BADWLRZ-SW13MB6.ads.mwn.de	"Gibt den FQDN des Servercomputers an, der die Authentifizierung der Nachricht für die Organisation ausgewertet hat." (Quelle: https://technet.microsoft.com/de-de/library/bb232136(v=exchg.150).aspx)
X-MS-Exchange-Organization-AuthAs: Anonymous	"Gibt die Authentifizierungsquelle an. Dieser X-Header ist immer vorhanden, wenn die Sicherheit einer Nachricht bewertet wurde. Die möglichen Werte sind Anonymous, Internal, External oder Partner." (Quelle: https://technet.microsoft.com/de-de/library/bb232136(v=exchg.150).aspx)

Aufgabe 5.2: DNS Server BIND

5.2.c)

"Nutzen sie das Programm dig um die Adresse www.wikipedia.org aufzulösen. Welche IP-Adresse steckt dahinter und wie viel Zeit nimmt die Auflösung in Anspruch?"

91.198.174.192, 14msec

Aufgabe 5.3: Rekursive und Iterative DNS Anfragen

"Gehen Sie von folgendem Szenario aus: Sie sitzen zu Hause auf dem Sofa und surfen mit Ihrem Tablet-PC, der über WLAN mit Ihrem DSL-Router verbunden ist, im Internet. Auf dem DSL-Router läuft ein eigener DNS-Server, welcher von Ihrem Tablet-PC als Standard-Nameserver genutzt wird. Gerade haben Sie die neue Adresse „www.gi.de“ im Browser Ihres Tablet-PC eingegeben und „OK“ zum Aufruf der Seite gedrückt."

5.3.a)

"Der DNS-Server auf Ihrem DSL Router bekommt somit eine DNS Anfrage von Ihrem Tablet PC. Stellen Sie den Aufbau der empfangenen DNS-Anfrage auf Transportschicht dar und geben Sie dabei alle verwendeten Felder der Paketköpfe (Header-Einträge) mit den enthaltenen Daten auf Transport- und Anwendungsschicht an."

Aufbau

Aufbau (wie Folie 2/58)	
Identification 0x99fa	Flags 0000 0 01 0 0
Number of questions 1	Number of answer RRs 0
Number of authority RRs 0	Number of additional RRs 0
Questions (variable number of questions) Queries www.gi.de: type A, class IN	

Answers (variable number of RRs)
Authority (variable number of RRs)
Additional information (variable number of RRs)

Header-Einträge

wireshark Feldname	Beschreibung / Protokollfeldname	Wert
Transportschicht		
udp.srcport	Source port: 36532 (36532)	36532
udp.dstport	Destination port: domain (53)	53
udp.port	Source or Destination Port: 36532	36532
udp.port	Source or Destination Port: 53	53
udp.length	Length: 35	35
udp.checksum_coverage	Checksum coverage: 35	35
udp.checksum	Checksum: 0x1846 [validation disabled]	0x1846
udp.checksum_good	Good Checksum: False	0
udp.checksum_bad	Bad Checksum: False	0
Anwendungsschicht		
dns.response_in	Response In: 17	17
dns.id	Transaction ID: 0x99fa	0x99fa
dns.flags	Flags: 0x0100 Standard query	0x0100
dns.flags.response	0... .. = Response: Message is a query	0
dns.flags.opcode	.000 0... .. = Opcode: Standard query (0)	0
dns.flags.truncated0. = Truncated: Message is not truncated	0
dns.flags.recdesired1 = Recursion desired: Do query recursively	1
dns.flags.z0.. = Z: reserved (0)	0
dns.flags.checkdisable0 = Non-authenticated data: Unacceptable	0
dns.count.queries	Questions: 1	1
dns.count.answers	Answer RRs: 0	0
dns.count.auth_rr	Authority RRs: 0	0
dns.count.add_rr	Additional RRs: 0	0
dns.qry.name	Name: www.gi.de	www.gi.de
dns.qry.type	Type: A (Host address)	0x0001

dns.qry.class	Class: IN (0x0001)	0x0001
---------------	--------------------	--------

5.3.b)

“Der Cache des DNS-Servers im Router und aller weiteren DNS-Server sei leer. Alle DNS Anfragen seien iterativ. Listen Sie alle bis zur erfolgreichen Auflösung notwendigen DNS Nachrichten mit Quelle und Ziel auf.”

Nachricht		Quelle	Ziel
Header	+-----+ OPCODE=QUERY QR=0, RD=1 +-----+	Tablet	NameServer (NS) DSL-Router
	+-----+	NameServer (NS) DSL-Router	NS Provider
Question	+-----+ QNAME=www.gi.de., QCLASS=IN, QTYPE=A +-----+	NS Provider	Root NS (liefert IP-Adresse an NS DSL-Router)
Answer	+-----+ <empty> +-----+		
Authority	+-----+ <empty> +-----+		
Additional	+-----+ <empty> +-----+		
Header	+-----+ OPCODE=QUERY, RESPONSE, AA QR=1, RD=1, RA=1 +-----+	DSL-Router	Tablet (erhält Response)
Question	+-----+ QNAME=www.gi.de., QCLASS=IN, QTYPE=A +-----+		
Answer	+-----+ www.gi.de. 1709 IN A 217.69.92.93 +-----+		
Authority	+-----+ <empty> +-----+		
Additional	+-----+ <empty> +-----+		

5.3.c)

“Betrachten Sie das gleiche Szenario wie in Aufgabenteil b), gehen Sie nun jedoch davon aus, dass alle Anfragen rekursiv gestellt sind und dass alle beteiligten DNS-Server rekursive Anfragen erlauben. Listen Sie wiederum alle bis zur erfolgreichen Auflösung notwendigen DNS Nachrichten mit Quelle und Ziel auf.”

Nachricht		Quelle	Ziel
Header	+-----+ OPCODE=QUERY QR=0, RD=1 +-----+	Tablet (bittet Standard-NS um vollständige Auflösung)	NS DSL-Router (kümmert sich um die vollständige Auflösung)
	+-----+		
Question	+-----+ QNAME=www.gi.de., QCLASS=IN, QTYPE=A +-----+		
Answer	+-----+ <empty> +-----+		
Authority	+-----+ <empty> +-----+		

Additional	+-----+ <empty> +-----+		
Header	+-----+ OPCODE=QUERY, RESPONSE, AA QR=1, RD=1, RA=1 +-----+	DSL-Router (liefert IP an Tablet)	Tablet (erhält Response)
Question	+-----+ QNAME=www.gi.de., QCLASS=IN, QTYPE=A +-----+		
Answer	+-----+ www.gi.de. 1709 IN A 217.69.92.93 +-----+		
Authority	+-----+ <empty> +-----+		
Additional	+-----+ <empty> +-----+		

6.1.1

“Als Paketgröße können Sie 1400 Byte nutzen. Warum wäre eine sehr viel geringere Paketgröße ungünstig?”

Weil die MTU (Maximum Transmission Unit) für Ethernet bei 1500 Bytes liegt. Pakete von bis zu 1500 Bytes können ohne Fragmentierung in den Rahmen (Frames) der Link-Schicht übertragen werden. liegt nun eine “sehr viel geringere” Paketgröße vor, bleiben große Teile des Frames ungenutzt. Da aber immer nur ganze Frames übertragen werden bleibt immer Platz im Frame übrig. Dadurch geht Geschwindigkeit verloren, weil deutlich mehr Daten pro Frame übertragen werden könnten.

Hinweise zu 6.1.a bis 6.1.c

k ist stets 250 msec. Die sonstigen Werte des Setups können den Tabellen bzw. Diagrammen entnommen werden. Bei der “Netzwerk”-Lösung in Teilaufgabe b ist der Server in der VM gelaufen, sonst auf dem gleichen Rechner.

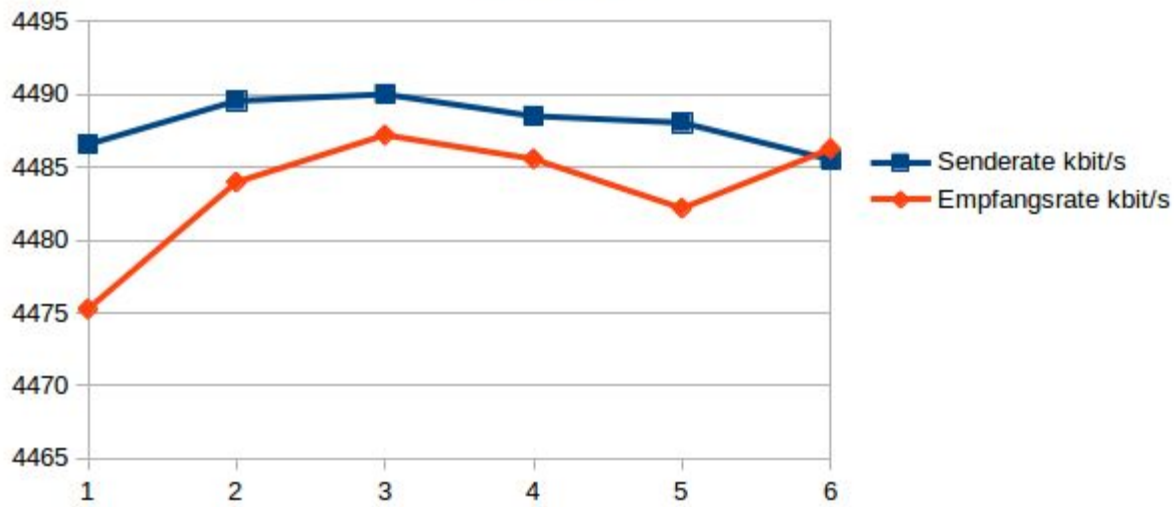
6.1.a UDP

Die erste Zeile entspricht den Werten N=5.000.000, die zweite N=5.000, die dritte N=100.

Minimum S	Minimum E	Mittelwert S	Mittelwert E	STABW S	STABW E
1218206,7008	1205443,353265	1406841,74354286	1389064,32555986	123630,4691156	122849,632451928
186582,371144	184871,587914	190856,678578833	187556,614963167	3471,5330356049	2201,6507654619
4485,515589	4475,259259	4488,0299838333	4483,4063071667	1,7253763282	4,3693321651

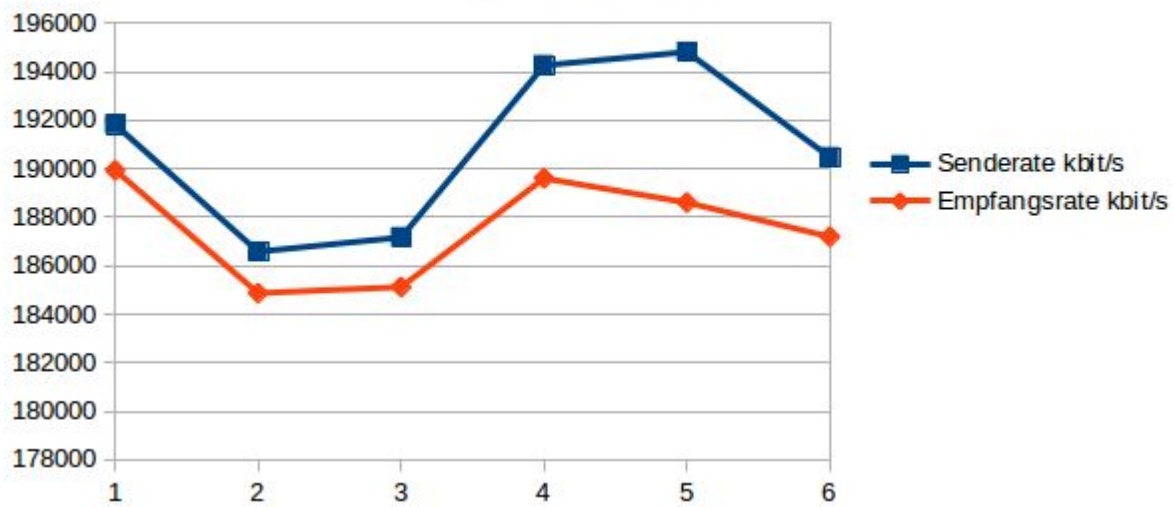
UDP local

k = 250, N = 100



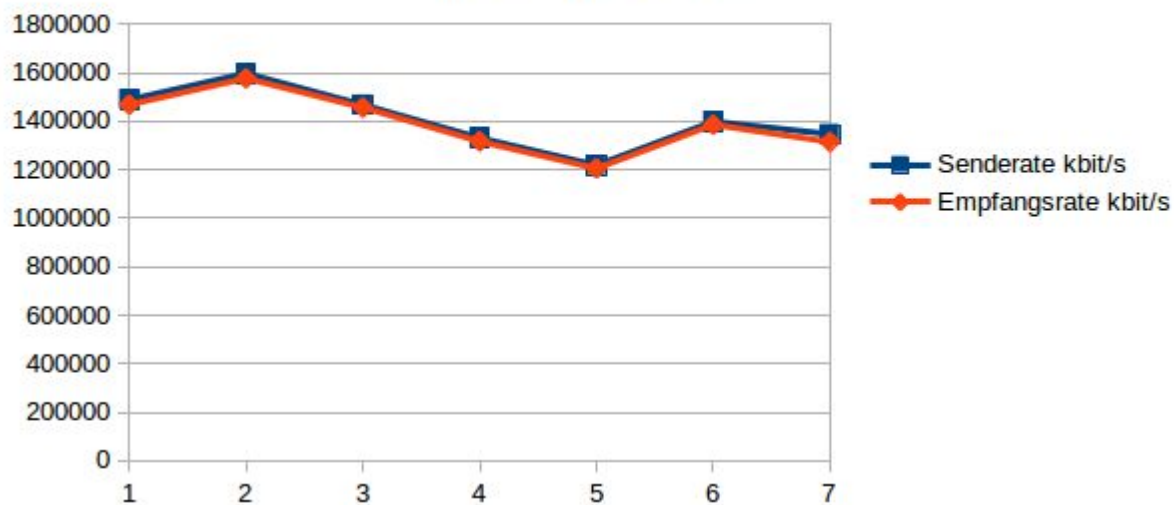
UDP local

k = 250, N = 5.000



UDP local

k = 250, N = 5.000.000



6.1.b Netzwerk

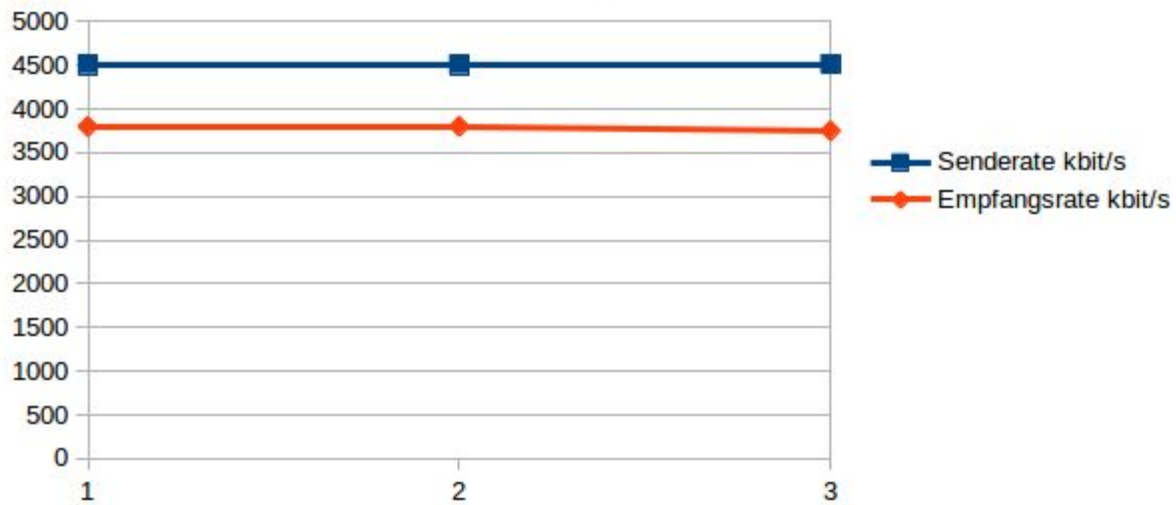
UDP

Die erste Zeile entspricht den Werten N=5.000.000, die zweite N=5.000, die dritte N=100.

Minimum S	Minimum E	Mittelwert S	Mittelwert E	STABW S	STABW E
1315534,6112	2721,512844	1366146,73066667	3440,0195623333	69392,8920348804	781,9970684988
195362,165031	3731,598668	196732,393205667	4020,028177	1186,6524109329	349,5440618344
4505,219899	3749,717808	4506,31559	3782,2479543333	1,7700553635	28,1855224494

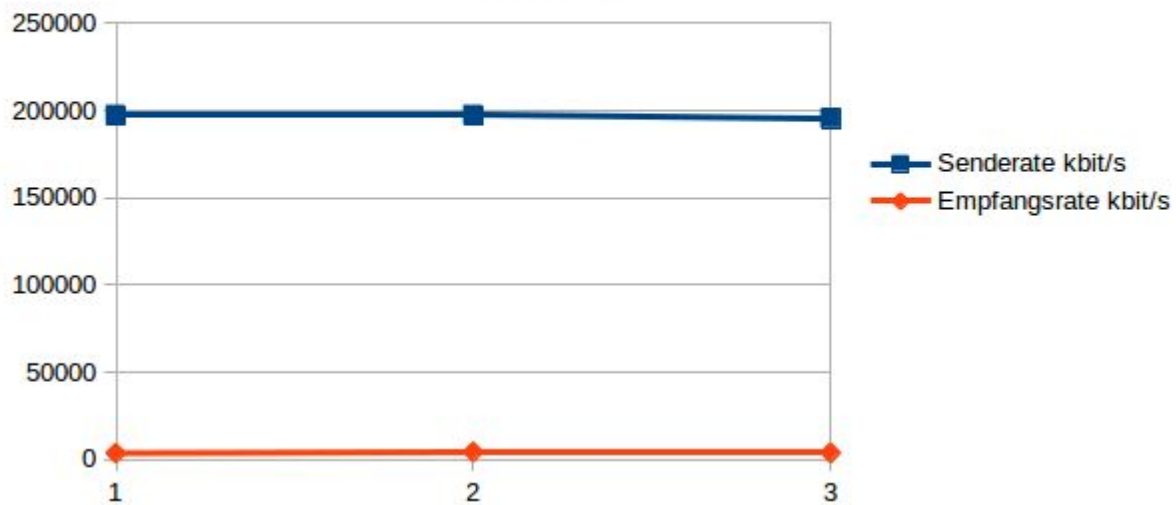
UDP Netzwerk

k=250, N=100



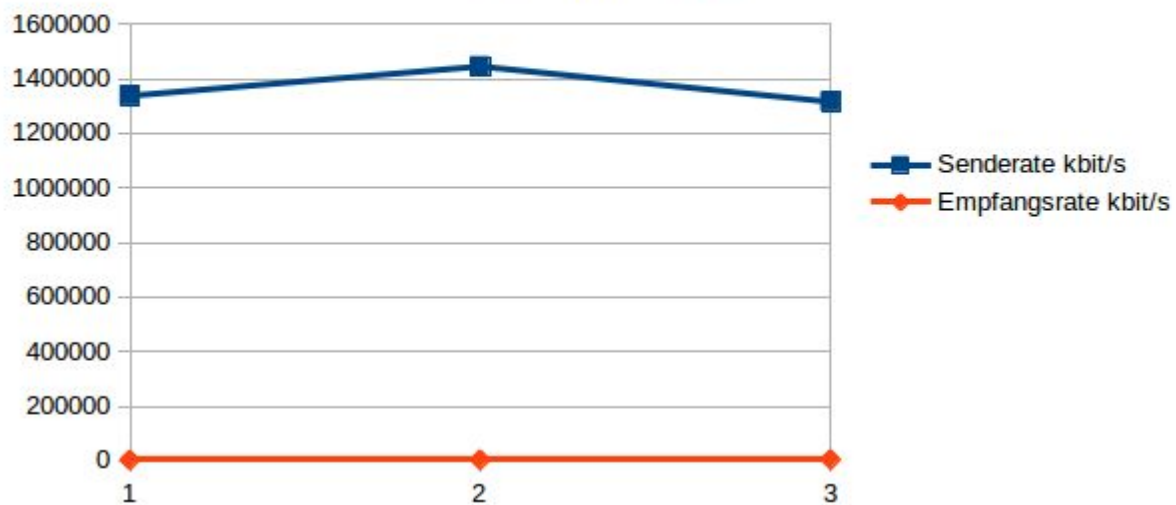
UDP Netzwerk

k=250, N=5.000



UDP Netzwerk

k=250, N=5.000.000



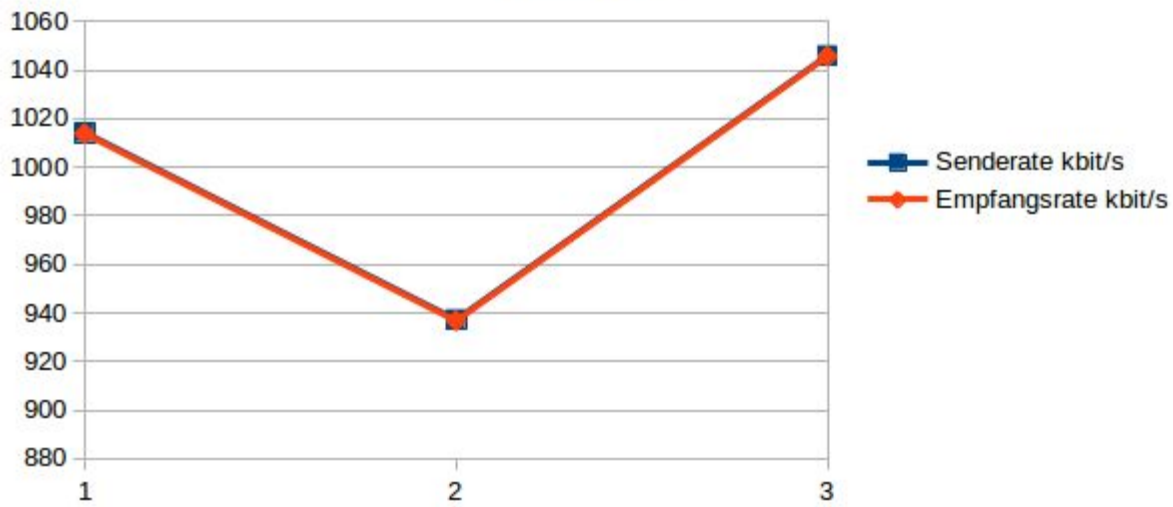
TCP

Die erste Zeile entspricht den Werten N=5.000.000, die zweite N=5.000, die dritte N=100.

Minimum S	Minimum E	Mittelwert S	Mittelwert E	STABW S	STABW E
847,107429	846,778626	856,1937323333	855,832168	11,2391295055	11,2051202532
812,455999	811,928293	834,903476	834,4599493333	20,4021529313	20,4351743579
937,22077	936,724771	999,1548356667	998,689541	55,9356009169	55,9737161784

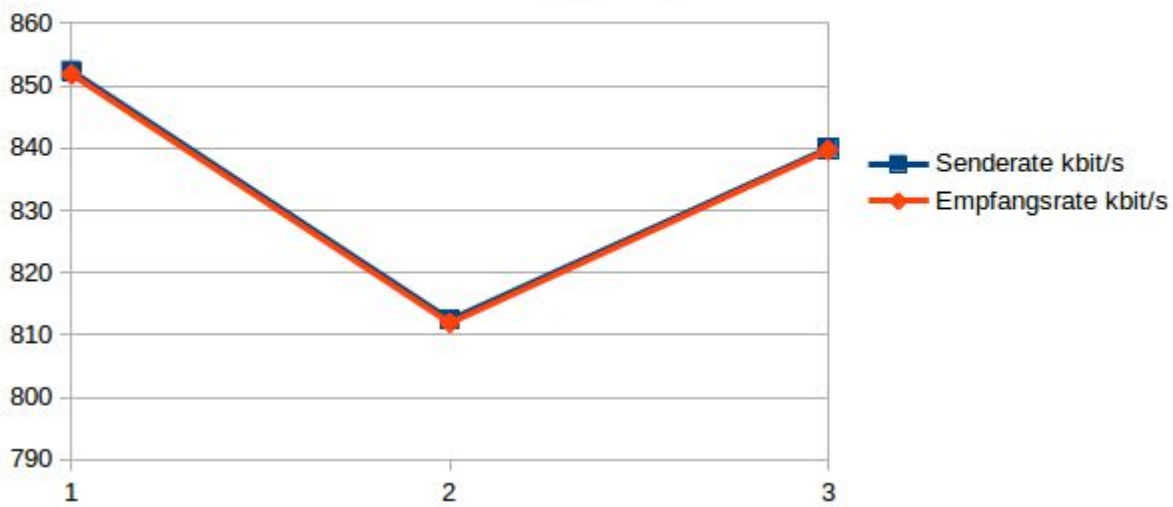
TCP Netzwerk

k=250, N=100



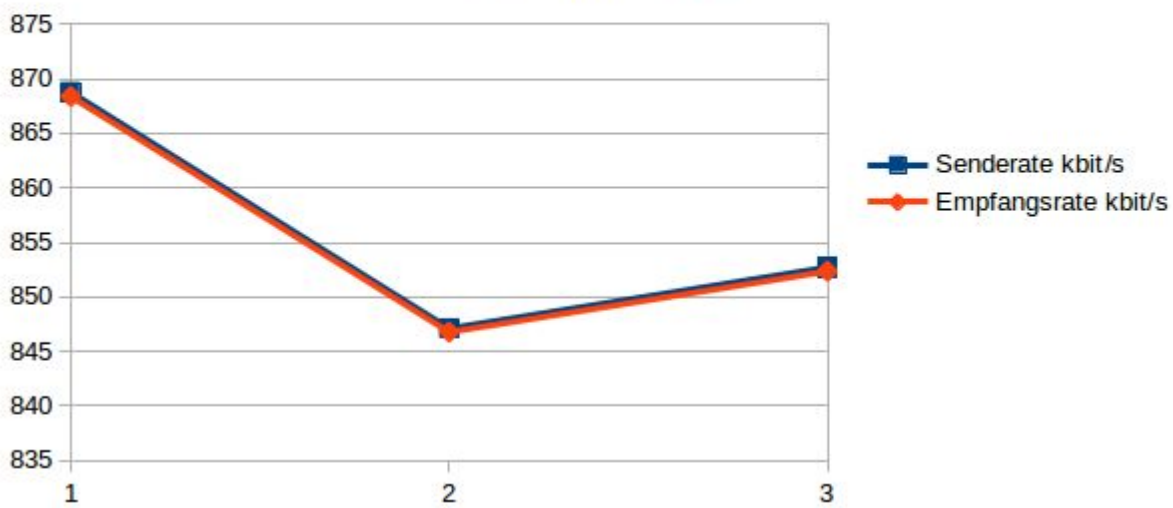
TCP Netzwerk

k=250, N=5.000



TCP Netzwerk

k=250, N=5.000.000



6.1.c TCP

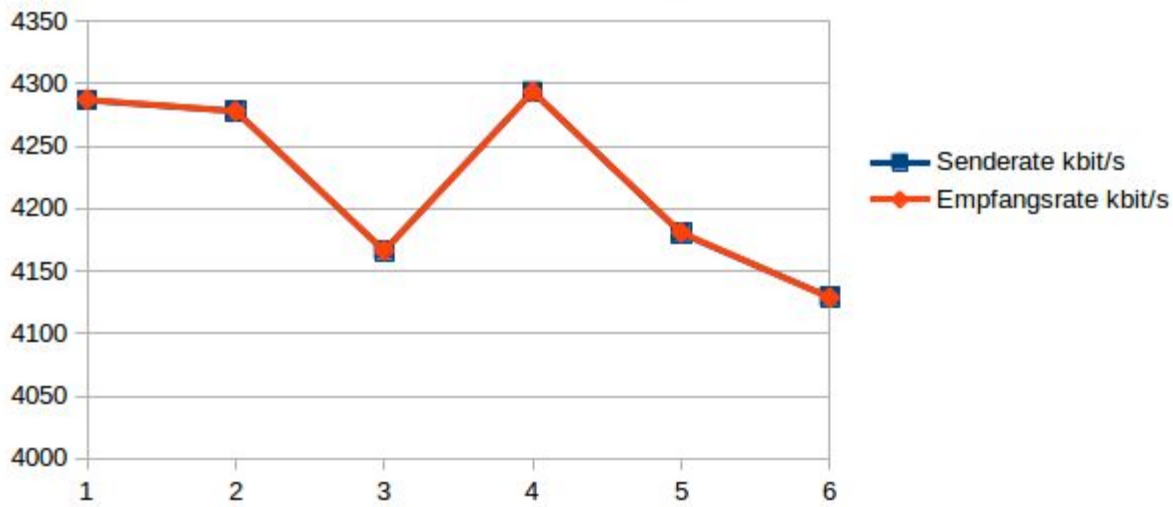
Die erste Zeile entspricht den Werten $N=5.000.000$, die zweite $N=5.000$, die dritte $N=100$.

Minimum S	Minimum E	Mittelwert S	Mittelwert E	STABW S	STABW E
10690,254858	57270,793503	10859,4897801667	62185,0881953333	174,7079158747	7443,2786664417
2789,7088	19197,1965	9038,9402026	37268,9372538	3382,9782633653	12349,664172504
4129,237499	4128,827038	4222,4583245	4222,529948	72,0379448249	72,128178477

Bei TCP sind Sende- und Empfangsrate im Netzwerk annähernd gleich.

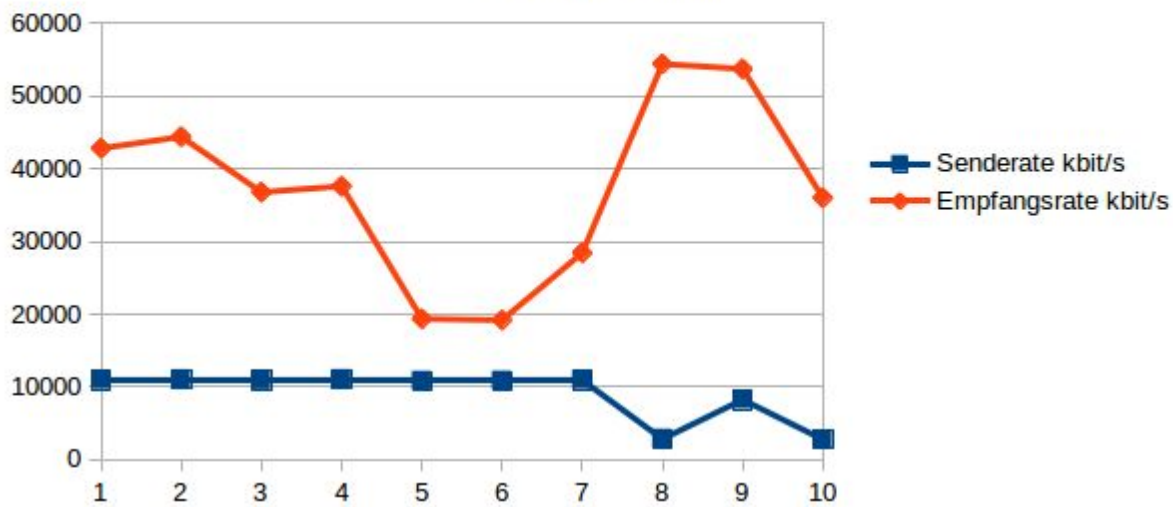
TCP local

k = 250, N = 100



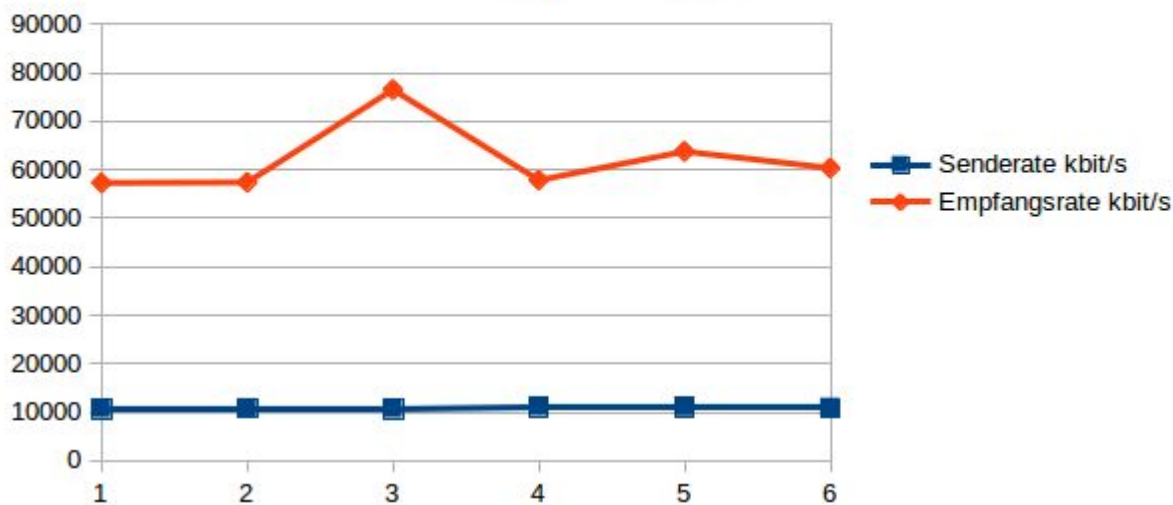
TCP local

k = 250, N = 5.000



TCP local

k = 250, N = 5.000.000



Aufgabe 6.2

Als eine Variante für die zuverlässige Datenübertragung haben Sie in der Vorlesung das Alternating Bit Protokoll/Stop-and-Wait Protokoll kennengelernt. Dieses soll in dieser Aufgabe theoretisch betrachtet werden. Eine Datei der Größe 4MB wird zwischen zwei Rechnern mittels Alternating Bit Protokoll übertragen. Zwischen dem Quell- und dem Zielrechner haben Sie eine Round-Trip-Time (RTT) von 50ms für ein Datenpaket und das dazugehörige ACK gemessen. Pro SnW Paket werden 1200 Byte an Nutzdaten übertragen.

6.2 a)

“Gehen Sie zunächst von einer Übertragung über einen zuverlässigen Kanal (keine Bitfehler, keine Paketverluste, kein Reordering) aus. Wie lange brauchen Sie, um die Datei zu übertragen?”

Annahme: 4 MB = $4 * 1000 * 1000$ Byte (Wegen Megabyte, keine Mebibyte)

Anzahl Pakete = $4\text{MB}/1200 \text{ Byte} = 3.333,3 = 3.334$ (aufrunden, weil es keine halben Pakete gibt)
Die RRT beträgt 50ms. Daraus folgt: Gesamtzeit = $3334 * 0,05\text{s} = 166,7 \text{ s}$

Es werden 166,7 Sekunden benötigt, um die Datei zu übertragen.

6.2 b)

“Der Kanal verwirft jetzt in der Richtung vom Quell- zum Zielrechner jedes zehnte Paket wegen Überlast. Die Rückrichtung ist nicht von Überlast betroffen. Paketverluste werden über einen Timer mit einer Dauer von 200ms detektiert und führen zu einer Neuübertragung des verlorenen Paketes. Wie lange brauchen Sie nun, um die Datei zu übertragen? (Nehmen Sie vereinfachend an, dass Neuübertragungen/Retransmissions nicht verloren gehen.)”

Benötigte Pakete: 3.334 (siehe a)

Anzahl Verlorener Pakete: $3334/10 = 333,4 = 333$ (abrunden, weil jedes 10 Paket verloren geht)

Wartezeit pro verlorenes Paket = 200ms

Gesamtzeit: $166,7\text{s} + (0,2\text{s} * 333) = 233,3\text{s}$

Es werden nun 233,3 Sekunden benötigt, um die Datei zu übertragen.

6.2 c)

Die Datenrate zwischen Quell- und Zielrechner beträgt 8 Mbit/s. Wie groß ist der Durchsatz, den Sie mit dem Alternating Bit Protokoll erreichen verglichen mit dem maximal möglichen Durchsatz? Wie könnten Sie – unter Beibehaltung der zuverlässigen Datenübertragung – den Durchsatz steigern?

Vergleich zu a)

Datenrate = $4\text{MB} * 8 / 166,7\text{s} = 0,19 \text{ MBit/s}$

Der Durchsatz beträgt im Vergleich zur Datenrate 2,38%

Vergleich zu b)

Datenrate = $4\text{MB} * 8 / 233,3\text{s} = 0,14 \text{ MBit/s}$

Der Durchsatz beträgt im Vergleich zur Datenrate 1,75%

Steigerung möglich durch Sliding Windows - Die Datenpakete werden gesendet, bevor ein ACK zurück kommt.