

# Internet Speed und das Kundenerlebnis

**Bachelorarbeit FS 2017**

Abteilung Informatik  
Hochschule für Technik Rapperswil

**Autoren:** Lukas Dürrenberger

Steven Ryser

**Betreuer:** Prof. Dr. Peter Heinzmann

**Experte:** Dr. Th. Siegenthaler, CSI

**Korreferent:** Prof. Dr. Daniel Patrick Politz

Ein spezieller Dank für die Unterstützung und  
das Korrekturlesen geht an folgende Leute:

Eric Franke

Rebecca Leuzinger

Felix Dürrenberger

Diane Dürrenberger

**Erstellt am: 20. März 2017**

**Letzte Änderung am: 16. Juni 2017**

## Abstract

---

Während früher noch alle Daten eines Videos offline verfügbar sein mussten, bevor man sie abspielen konnte, hat mittlerweile Streaming den Videokonsum revolutioniert. Heutzutage werden Internet-Heimanschlüsse hauptsächlich mit dem Qualitätsmass «Datenrate» verkauft. Aktuell werden Datenraten von bis zu 1 Gbit/s angeboten. Der Bundesrat hat kürzlich die minimale Datenübertragungsrate für den Internetzugang per 2018 auf 3 Mbit/s Download festgelegt.

Im Rahmen dieser Bachelorarbeit sollen Parameter identifiziert und erfasst werden, welche das Kundenerlebnis bzw. die Videoqualität beim Videostreaming bestimmen. In diesem Zusammenhang werden auch verschiedene Videostandards und Videostreamingportale verglichen und die besten Portale evaluiert. Die Big Data Anwendung Splunk indexiert, analysiert und visualisiert die erfassten Qualitätsdaten. Diese Anwendungen von Splunk dienen als Grundlagen für einen «Big Data Weiterbildungskurs» an der HSR.

Verschiedene Videoportale wurden gemäss Benutzer- und Bekanntheitsstatistiken gefiltert und evaluiert. Diese Portale wurden analysiert, um herauszufinden, welche Parameter das Kundenerlebnis beeinflussen. Im Anschluss an die Bestimmung verschiedener Parameter, wie zum Beispiel Protokolltyp, Auflösung oder Segmentgrösse, wurde eine Anwendung in Python erstellt, welche die momentanen Videostreams laufend analysiert, die vorher bestimmten Parameterdaten herausfiltert und in einer Datei festhält. Diese Messdaten wurden durch einen Monitor in Splunk hochgeladen und indexiert. Im weiteren Verlauf wurden damit Auswertungen mit der Splunk-eigenen Suchsprache und Visualisierungen erstellt. Aus den Auswertungen wurde ersichtlich, dass Teleboy die qualitativsten Streams für Gratisbenutzer hat. Im Hinblick auf den Splunk-Kurs stellte sich die Option mit einem ZAP Proxy als die Beste heraus. Der Inhalt dieses Splunk-Kurses wurde definiert, mit dem Ziel Splunk kennenzulernen und damit verschiedene Daten zu indexieren, analysieren und visualisieren. Zusätzlich wurde ein Debian-Image erstellt, welches als Proxy funktioniert, so dass wissbegierige Kursteilnehmer dieses auf einem dafür bestimmten Computer vorübergehend mit nach Hause nehmen können, um den Kursinhalt zu wiederholen und intensiver in die Materie einzutauchen.

## Aufgabenstellung

---



### Studien- oder Bachelorarbeit «Internet Speed und das Kundenerlebnis»

---

Studiengang: Informatik (I)

---

Institut: ITA: Internet-Techn. und Anwendungen  
Gruppe: Lukas Dürrenberger und Steven Ryser

---

Betreuer: Heinzmann, Peter  
Koreferent: Prof. Dr. Daniel Patrick Politze, HSR  
Experte: Dr. Th. Siegenthaler, CSI  
Industriepartner:

### Ausgangslage

---

Gegenwärtig werden Internet Heimanschlüsse vor allem mit dem Qualitätsmass «Datenrate» angepriesen. Swisscom und UPC verkaufen den Heimanwendern 250 Mbit/s oder sogar 1 Gbit/s. Die European Commission will mit ihrer Broadband Europe Strategie bis 2025 mindestens 100 Mbit/s für jeden Haushalt anbieten. Der Schweizerische Bundesrat hat 2016 die Revision der Verordnung über Fernmeldedienste (FDV) verabschiedet und dabei die minimale Datenübertragungsrate für den Internetzugang per 2018 auf 3 Mbit/s Download und 300 Kbits/s Upload Datenrate festgelegt. Diese Übertragungsraten würden gemäss dem Bundesrat der voraussichtlichen Evolution entsprechen und sie seien ausreichend, um sämtliche Grundversorgungsdienste in guter Qualität zu gewährleisten.

Es gibt offensichtlich ziemliche Diskrepanzen zwischen den Datenraten, welche verschiedene Organisationen für die künftigen Internet Anschlüsse als ausreichend ansehen. Aktuelle Untersuchungen zeigen, dass das «Kundenerlebnis» bzw. die «User Experience» im Internet nicht nur von der Datenrate abhängt. So können beispielsweise die Antwortzeiten beim Surfen mit der Erhöhung der Datenraten auf über 20 Mbit/s nicht mehr wesentlich verbessert werden. Auch Video- und Audiodienste benötigen kaum mehr als 20 Mbit/s pro Nutzer. Langsame Antwortzeiten sind bei den Anschlussdatenraten von mehr als 20 Mbit/s häufig auf Probleme im lokalen Netz (WLAN, Powerline), auf DNS-Probleme oder suboptimal programmierte Webseiten zurückzuführen.

Im Rahmen dieser Studienarbeit sollen neue Messkriterien zur Qualität der Internetdienste bzw. der User Experience untersucht werden. Es sind Anwendungen zu entwickeln, welche die Messung und Auswertung von neuen User Experience Kriterien ermöglichen. Dabei ist abzuklären, wie Data Analytics Werkzeuge (Splunk) für die Charakterisierung, Erfassung und Auswertung der User Experience genutzt werden können.

## Ziel

---

Es soll eine Mess- und Auswerteanwendungen zu Verfügung stehen, welche quantitative Aussagen über die Qualität bei Video Streaming Diensten erlaubt. Die Datenerfassung, Analyse und Visualisierung soll mit Hilfe von Splunk erfolgen. Diese Grundlagen sollen für die Präsentation der User Experience Grundlagen beim Video Streaming und für die Durchführung eines Splunk Kurses für Studierende genutzt werden können. Im Rahmen der Arbeit sind auch Vorschläge für die Durchführung eines Splunk Kurses für Studierende auszuarbeiten.

## Aufgaben

---

Die konkreten Teilaufgaben werden an die Vorkenntnisse und Präferenzen der Studenten angepasst. Unter anderem stehen folgende Teilaufgaben zur Diskussion.

1. Einarbeitung:
  - a. Definition, Beschreibung der Begriffe (z.B. User Experience, Real User Monitoring, Application Performance Management, Passive Monitoring)
  - b. Studium verschiedener Methoden und Lösungen zum User Monitoring beim Video Streaming
  - c. Studium verschiedener Methoden zur Charakterisierung der «User Experience» beim Video Streaming
  - d. Erfassung und Analyse typischer Datenraten und Datenmengen bei verschiedenen beim Video Streaming Anwendungen
  - e. Analyse und Visualisierung von User Experience Daten mit Big Data Ansätzen (z.B. Splunk User Behavior Analytics)
2. Technische Möglichkeiten zur Erfassung des Benutzerverhaltens
  - a. Erfassung der aktivierten Video Streaming Anwendungen
  - b. Erfassung von User Experience Faktoren auf Server- und/oder auf Clientseite (Agenten)
3. Definition und Realisierung eines Splunk Prototypen
  - a. Datenerfassung mittels Forwarder auf Speedtest Mess-Sonden (Raspberry Pi), Mobiles, Firewalls
  - b. Analyse, statistische Auswertungen (Korrelationen)
  - c. Quantifizierung und Visualisierung des Benutzerverhaltens und der User Experience Parameter

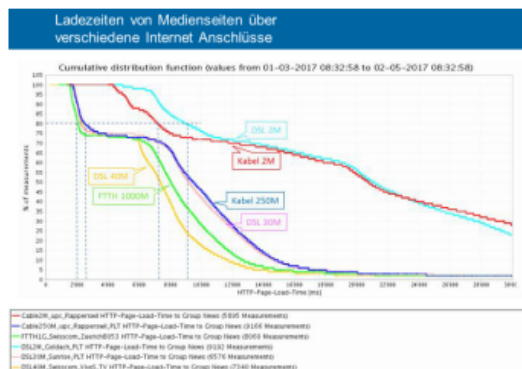
## Referenzen

---

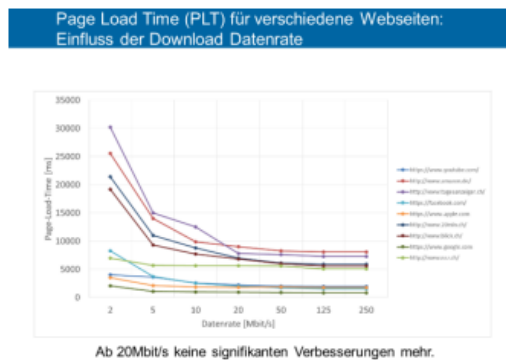
1. Hinweise zur Durchführung von Studienarbeiten: [https://dl.dropboxusercontent.com/u/4679041/SABA-Web-Anleitungen\\_Heinzmann.zip](https://dl.dropboxusercontent.com/u/4679041/SABA-Web-Anleitungen_Heinzmann.zip)
2. [https://de.wikipedia.org/wiki/User\\_Experience](https://de.wikipedia.org/wiki/User_Experience)
3. [https://en.wikipedia.org/wiki/Real\\_user\\_monitoring](https://en.wikipedia.org/wiki/Real_user_monitoring)
4. [https://en.wikipedia.org/wiki/Application\\_performance\\_management](https://en.wikipedia.org/wiki/Application_performance_management)
5. Splunk <https://de.wikipedia.org/wiki/Splunk>
6. Client Side Splunk, <http://blogs.splunk.com/2013/02/07/client-side-splunk/>

7. Splunk User Behavior Analytics: The Whiteboard (Security Application)  
[https://www.splunk.com/en\\_us/resources/video.A4Z2ZjNTE6yjeR92MB0yji\\_79eLZueOh.html](https://www.splunk.com/en_us/resources/video.A4Z2ZjNTE6yjeR92MB0yji_79eLZueOh.html)
8. Google Page Speed Infos <https://developers.google.com/speed/docs/insights/about>
9. Sonia Burney, Sabrina Burney, Security and Frontend Performance, O'Reilly, January 2017.
10. Web Performance Today <http://www.webperformancetoday.com/2015/04/15/new-findings-state-union-ecommerce-page-speed-web-performance-spring-2015/>
11. Google Testing A Red "Slow" Label in Search Results for Slower Sites  
<http://searchengineland.com/google-testing-red-slow-label-search-results-slower-sites-215483>
12. Splunk for Customer Experience, User Analytics  
[https://www.splunk.com/en\\_us/solutions/solution-areas/business-analytics/customer-experience.html](https://www.splunk.com/en_us/solutions/solution-areas/business-analytics/customer-experience.html)
13. Dive into Splunk Lehrgang <http://www.innovato.com/splunk/>

## Weitere Informationen



Gemessene Webseiten Ladezeiten (Web Page Load Time, PLT) für verschiedene Medienseiten beim Zugriff via verschiedene Internet Zugänge



Die Ladezeiten von den meisten Webseiten verbessern sich ab etwa 20 Mbit/s Download Datenrate nicht mehr signifikant.

Notizen zu einem möglichen Blockkurs an der HSR Hochschule für Technik Rapperswil.

Arbeitstitel: Performance vs. User Experience im Internet und Big Data

Industriepartner: Splunk, cnlab, Swisscom

Methodik: Projektbasiertes Lernen, Arbeiten in zweier Teams und Integration von allen Ergebnissen der Teilnehmenden zu einem Dashboard

Lernkontrolle/Prüfung: Lernkontrolle ([www.mobilequiz.ch](http://www.mobilequiz.ch)) zu den Wissensfragen, Praktische Aufgaben mit Splunk, Theorieprüfung

Vorkenntnisse: Studierende des Studiengangs Informatik, welche mindestens drei Semester abgeschlossen haben. Sie müssen Computernetze 1 und Informationssicherheit 1 bereits bestanden haben.

Kurzbeschreibung:

Mobil- und Computernetzanbieter preisen Privatkunden und Mobilnutzern immer höhere Datenraten an. Höhere Datenraten garantieren aber nicht immer bessere «User Experience». Wie man «User Experience» beschreibt und quantitativ erfasst, ist nicht offensichtlich. Eine Vielzahl von Faktoren bestimmen die User Experience bei verschiedenen Anwendungen (Surfing, Streaming, Chatting), Personen (Anwendertypen), Situationen (Büro, Heim, Unterwegs), Computing Plattformen (Desktop, Notebook, Tablet, Smartphone) und Infrastrukturen (Wireline, Mobilnetz, Content Distribution Server). Für so eine komplexe Situation mit unzähligen beeinflussenden Faktoren bieten sich Big Data und Analytics Ansätze an, um wichtige User Experience Indikatoren erfassen und visualisieren zu können. In diesem Kurs lernen die Teilnehmenden, wie man anhand von Experimenten mit Netzwerkgeräten (Router, Modem, Probes, Agents), Notebooks, Tablets und Smartphones in Ergänzung zu Internet Performance Messungen auch die User Experience erfassen, analysieren und visualisieren kann. Die Teilnehmenden erfahren, was unter «User Experience» bei Internet Anwendungen zu verstehen ist. Im Rahmen eines Pilotprojekts lernen die Teilnehmenden die wichtigsten Komponenten und Prozesse von Big Data Anwendungen kennen. Mit Hilfe von Splunk realisieren sie Grundbausteine welche schliesslich zu einem «User Experience» Dashboard mit Live Visualisierung zu Key Performance Indikatoren zusammengesetzt werden können.

Notizen, Stichworte

- Analysis of web application logs
- We want to see in real-time how people use applications: Mit Splunk Version 5.0 bekommen Unternehmen umfassende Visibilität und können so Performance-Probleme besser identifizieren und Ausfälle verhindern [http://www.splunk.com/de\\_de/view/SP-CAAHTS](http://www.splunk.com/de_de/view/SP-CAAHTS)
- [https://www.splunk.com/en\\_us/products/premium-solutions/user-behavior-analytics.html](https://www.splunk.com/en_us/products/premium-solutions/user-behavior-analytics.html)
- <http://www.fhnw.ch/aps/weiterbildung/cas/cas-usability-user-experience>
- <https://www.digicomp.ch/weiterbildung/softwareentwicklungs-trainings/software-engineering/software-ergonomie-und-usability/einfuehrung-in-usability-und-user-experience-ux>

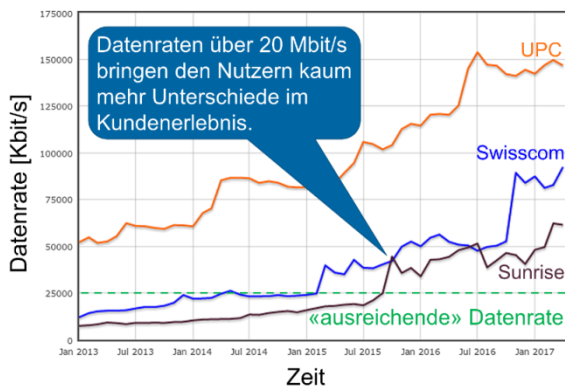
Notizen

- Android App Erweiterungen
  - Ping
  - Youtube Details (Nerd Outputs) auslesen
- User Experience Objekte klassieren
  - Ads
  - Tracker
  - Content
  - Anteile von Datenmengen und Objekten messen (Zwischenzeiten bei der Webseiten Darstellung)

## Management Summary

### Ausgangslage:

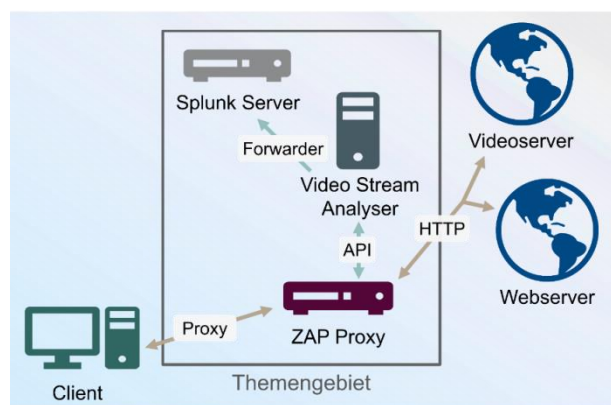
Videostreaming ist ein Begriff, der immer bekannter wird. Während früher noch alle Daten eines Videos offline verfügbar sein mussten, so wird heute das Streaming immer populärer und bietet ganz neue Möglichkeiten zum Videokonsum, unter der Voraussetzung, dass das Internet zugänglich ist. Heutzutage werden Internet-Heimanschlüsse hauptsächlich mit dem Qualitätsmass «Datenrate» verkauft. Aktuell werden Datenraten von bis zu 1 Gbit/s angeboten. Der Bundesrat hat kürzlich die minimale Datenübertragungsrate für den Internetzugang per 2018 auf 3 Mbit/s Download und 300 Kbit/s Upload festgelegt. Gemäss dem Bundesrat würden diese Übertragungsraten der voraussichtlichen technischen Evolution entsprechen und seien ausreichend, um sämtliche Grundversorgungsdienste in guter Qualität zu gewährleisten. Damit gibt es offensichtliche Meinungsdivergenzen zum Thema der Wichtigkeit von hohen Datenübertragungsraten.



Frühere Untersuchungen von Ladezeiten von Webseiten zeigen, dass das Kundenerlebnis im Internet nicht nur von der Datenrate abhängt. So können beispielsweise Antwortzeiten beim Surfen ab 20 Mbit/s nicht mehr verbessert werden. Somit wird ersichtlich, dass noch weitere, möglicherweise weniger bekannte, Parameter eine Rolle spielen.

### Vorgehen/Technologien

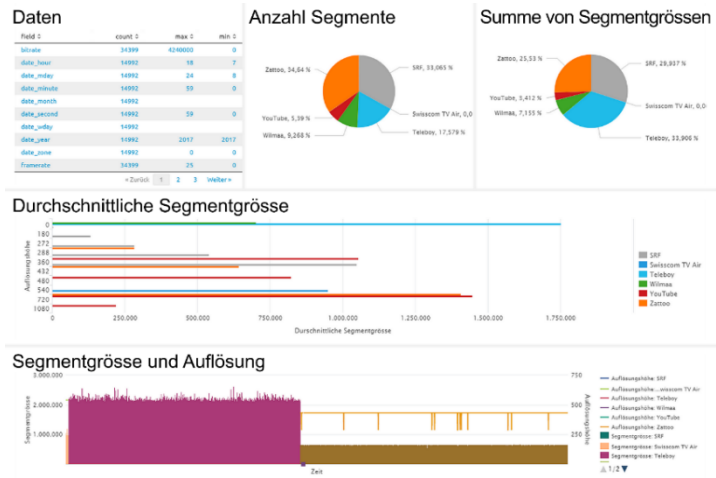
Im Rahmen dieser Bachelorarbeit sollen Parameter identifiziert und erfasst werden, welche das Kundenerlebnis bzw. die Videoqualität beim Videostreaming bestimmen. In diesem Zusammenhang werden auch verschiedene Videostandards und Videostreamingportale verglichen und die besten Portale evaluiert. Dazu soll eine neue Applikation zur Auswertung in Python programmiert werden, welche die Parameterwerte automatisch speichert. Die Big Data Anwendung Splunk indexiert, analysiert und visualisiert die somit erfassten Qualitätsdaten. Dadurch soll eine Aussage über die Qualität des entsprechenden Videoportals ermöglicht und mit Auswertungen anderer Videoportale verglichen werden. Diese Anwendungen von Splunk dienen als Grundlagen für einen «Big Data Weiterbildungskurs» an der HSR, wobei die Studenten die Möglichkeit haben, Teile dieses Kurses zuhause zu wiederholen.





### Ergebnis

Verschiedene Videoportale wurden gemäss Benutzer- und Bekanntheitsstatistiken gefiltert und evaluiert. Diese Portale wurden analysiert, um herauszufinden, welche Parameter das Kundenerlebnis beeinflussen. Im Anschluss an die Bestimmung der verschiedenen Parameter, wie zum Beispiel Protokolltyp, Auflösung oder Segmentgrösse, wurde eine Anwendung in Python erstellt, welche die momentanen Videostreams laufend analysiert, die vorher bestimmten Parameterdaten herausfiltert und in einer Datei festhält. Diese Messdaten wurden dann durch einen Monitor in Splunk hochgeladen und indexiert. Im weiteren Verlauf wurden damit Auswertungen mit der Splunk-eigenen Suchsprache und Visualisierungen erstellt. Aus den Auswertungen wurde ersichtlich, dass Teleboy die qualitativsten Streams für Gratisbenutzer hat. Im Hinblick auf den Splunk-Kurs stellte sich die Option mit einem ZAP Proxy als die Beste heraus. Der Inhalt dieses Splunk-Kurses wurde definiert, mit dem Ziel Splunk kennenzulernen und damit verschiedene Daten zu indexieren, analysieren und visualisieren. Zusätzlich wurde ein Debian-Image erstellt, welches als Proxy funktioniert, so dass wissbegierige Kursteilnehmer dieses auf einem geeigneten Computer vorübergehend mit nach Hause nehmen können, um den Kursinhalt zu wiederholen und intensiver in die Materie einzutauchen.



## Inhaltsverzeichnis

---

<b>Abstract</b> .....	<b>2</b>
<b>Aufgabenstellung</b> .....	<b>3</b>
<b>Management Summary</b> .....	<b>7</b>
Ausgangslage: .....	7
Vorgehen/Technologien .....	7
Ergebnis.....	8
<b>Inhaltsverzeichnis</b> .....	<b>9</b>
<b>1 Einführung</b> .....	<b>12</b>
1.1 Ausgangslage .....	12
1.2 Ziel.....	13
1.3 Rahmenbedingungen.....	13
<b>2 Technische Grundlagen</b> .....	<b>14</b>
2.1 Videoformate .....	14
2.1.1 Allgemeiner Prozess .....	14
2.1.2 Codec .....	14
2.1.3 Containerformat .....	15
2.2 Videostreaming.....	16
2.2.1 Videostream und Segmente.....	16
2.2.2 Buffering .....	17
2.2.3 Replay und Live Videos.....	18
2.2.4 Unterbrechungen .....	18
2.2.5 Transport Protokoll.....	19
2.2.6 Streaming Protokoll.....	19
2.3 Apposite Linktropy 5500.....	21
2.3.1 Einstellungen .....	22
2.3.2 Scheduler.....	23
2.4 OWASP Zed Attack Proxy (ZAP) .....	25
2.4.1 HTTPS Verkehr.....	25
2.4.2 Application Programming Interface (API).....	25
2.5 Splunk.....	26
2.5.1 Erfassung.....	26
2.5.2 Auswertung.....	26
2.5.3 Visualisierung.....	27
<b>3 Analyse von Videoportalen</b> .....	<b>28</b>
3.1 Einführung .....	28
3.1.1 Ziel.....	28

---

3.1.2	Vorgehen .....	28
3.2	Videoportale.....	28
3.2.1	USA .....	28
3.2.2	Deutschland .....	29
3.2.3	Schweiz .....	30
3.2.4	Providers.....	31
3.3	Technischer Aufbau .....	32
3.3.1	Aufbau .....	32
3.3.2	Vorbereitung .....	32
3.3.3	Erfassung.....	33
3.4	Messungen .....	34
3.4.1	Qualität .....	34
3.4.2	Paketverlust .....	38
3.4.3	Werbung .....	42
<b>4</b>	<b>Entwicklung eines Video Stream Analysers .....</b>	<b>43</b>
4.1	Technischer Aufbau .....	43
4.2	Struktur .....	44
4.3	Funktionen .....	45
4.3.1	Service Detection .....	45
4.3.2	Service Tracking .....	45
4.3.3	Merkmale .....	46
<b>5</b>	<b>Auswertung mit Splunk .....</b>	<b>47</b>
5.1	Splunk Möglichkeiten .....	47
5.2	Erfassung.....	47
5.2.1	Manuelles Hochladen.....	47
5.2.2	Quelle überwachen .....	48
5.2.3	Weiterleiten .....	48
5.3	Auswertung .....	49
5.4	Visualisierung .....	51
5.5	Splunk Kurs .....	52
5.5.1	Kursinhalt .....	52
5.5.2	Use Cases .....	52
5.5.3	Kursutensilien .....	55
<b>6</b>	<b>Auswertungen und Ergebnisse.....</b>	<b>56</b>
6.1	Auswertungen mit Splunk .....	56
6.2	Videostream Provider.....	57
6.2.1	Teleboy .....	58
6.2.2	Zattoo.....	58
6.2.3	YouTube .....	59

6.2.4	Swisscom TV Air .....	59
6.2.5	Wilmaa .....	60
6.2.6	SRF .....	60
<b>7</b>	<b>Schlussfolgerung</b> .....	<b>61</b>
7.1	Zusammenfassung .....	61
7.2	Ausblick .....	61
<b>8</b>	<b>Anhang</b> .....	<b>62</b>
8.1	Verzeichnisse .....	62
8.1.1	Literatur- und Quellenverzeichnis .....	62
8.1.2	Glossar und Abkürzungsverzeichnis .....	64
8.1.3	Abbildungen .....	65
8.1.4	Tabellen .....	66
8.1.5	Formeln .....	66
8.2	Projektmanagement .....	67
8.2.1	Projektplan .....	67
8.2.2	Zeitauswertung .....	68
8.3	Persönliche Berichte .....	71
8.3.1	Lukas Dürrenberger .....	71
8.3.2	Steven Ryser .....	71
8.4	Archivdatei Inhalt .....	72

# 1 Einführung

## 1.1 Ausgangslage

Internetanschlüsse werden gegenwärtig fast ausschliesslich über die Datenübertragungsraten vermarktet. Dass die Übertragungsrate jedoch nicht das einzige ausschlaggebende Kriterium für ein gutes Kundenerlebnis im Internet ist, konnte bereits durch Auswertungen von Seitenladezeiten durch die cnlab festgestellt werden [1].

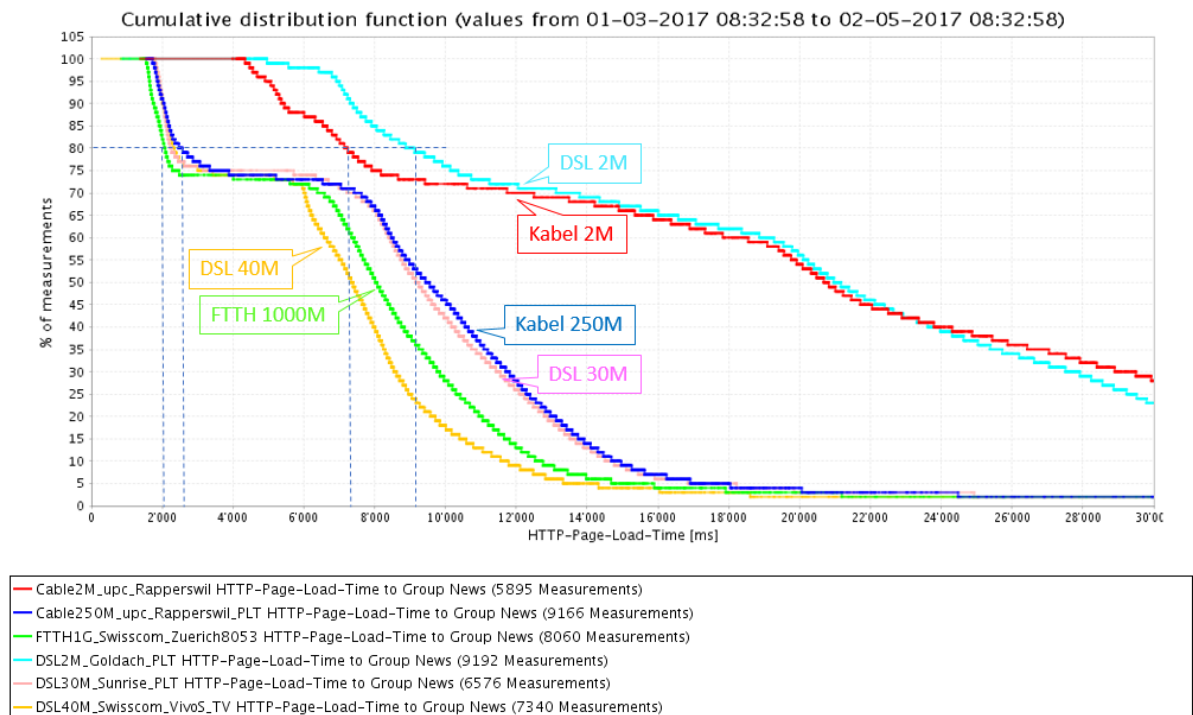


Abbildung 1: Seitenladezeit bei verschiedenen Anbietern

Das Beispiel in Abbildung 1 zeigt die Verteilung der Ladezeiten bei einer Gruppe von Newsseiten mit verschiedenen Downloadraten von 2, 30, 40, 250 und 1000 Mbit/s. Der Ladezeitenunterschied ist trotz grosser Differenzen in den Downloadraten nur minimal. Es lässt sich daraus schliessen, dass ab einer bestimmten Datenübertragungsrate andere Kriterien an Wichtigkeit gewinnen müssen, wenn es darum geht Ladezeiten zu minimieren um dem Kunden ein besseres Erlebnis zu bieten.

Dieses Wissen, dass die Datenrate nicht das einzige ausschlaggebende Kriterium für ein gutes Kundenerlebnis bei Webseiten ist, dient als Grundlage dieser Arbeit für die Suche nach messbaren Kriterien bei Videostreams von verschiedenen Videoportalen.

### 1.2 Ziel

---

Es soll eine Messanwendung entwickelt werden, welche gute Aussagen über die Qualität beim Konsum von Videostreams auf verschiedenen Geräten wie Smartphone oder Notebook erlaubt. Nebst dem Erfassen der Daten sollen diese zusätzlich mit Splunk analysiert und visualisiert werden.

In einem ersten Schritt sollen verschiedene Videoportale analysiert werden, um herauszufinden welche technischen Merkmale es gibt, um das Kundenerlebnis zu erfassen.

In einem weiteren Schritt soll dann geprüft werden wie diese Merkmale mit einer eigenen Anwendung automatisch erkannt und aggregiert werden können.

Schlussendlich sollen die erfassten Daten an Splunk weitergegeben und dort mit spezifischen Suchtermen analysiert und visualisiert werden.

### 1.3 Rahmenbedingungen

---

Während der Analysephase wurde festgestellt, dass das Analysieren und Auswerten von Videoportalen einen sehr grossen Aufwand darstellt und dass die Erweiterung auf andere Internetdienste nur beschränkt möglich wäre, weshalb das Themengebiet ausschliesslich auf Videostreams und Videoportale beschränkt wurde.

## 2 Technische Grundlagen

### 2.1 Videoformate

Die Arbeit beschäftigt sich stark mit Videodaten und es werden verschiedenste technische Begriffe gebraucht. Dieser Abschnitt soll einige dieser Begriffe im Voraus erklären, um möglichen Missverständnissen vorzubeugen.

#### 2.1.1 Allgemeiner Prozess

Ein Video ist nichts anderes als eine schnelle Abfolge von Einzelbildern (Frames). Ab einer Framerate von 16 Frames pro Sekunde (FPS) empfindet es der Mensch als ein fließendes Bild und nimmt die einzelnen Aufnahmen nicht mehr bewusst wahr [2].

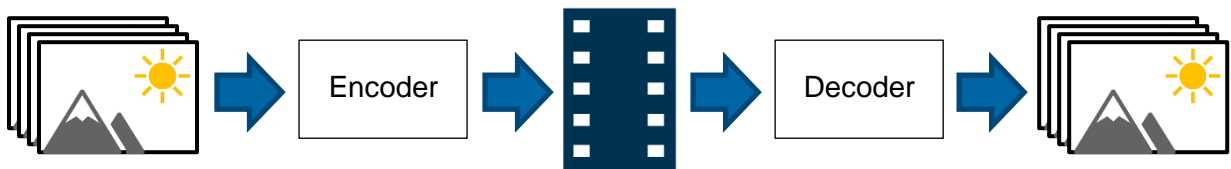


Abbildung 2: Allgemeiner Prozess von Videos

Wie in Abbildung 2 gezeigt, werden die Einzelbilder mittels eines Encoders so in ein Videoformat codiert und vor dem Betrachten durch einen Decoder wird wieder in einzelne Frames decodiert, mit dem Ziel, dass die zu übertragende oder abzuspeichernde Datenmenge möglichst klein wird ohne die Bildqualität merklich zu beeinträchtigen.

#### 2.1.2 Codec

Ein Videocodec bezeichnet ein Algorithmen-Paar zur Kodierung (Encoder) und Dekodierung (Decoder) von digitalen Videos für die Übertragung oder Speicherung.

Die Codierung kann verlustbehaftet (lossy) oder verlustfrei (lossless) sein. Da verlustfreie Komprimierungen zu enormen Dateigrößen führen, wird für Videostreams das Videomaterial in der Regel mit verlustbehafteten Algorithmen komprimiert.

Bei der Komprimierung wird zwischen dem Komprimieren eines gesamten Frames (Intraframe-Komprimierung) und dem Komprimieren zwischen zwei oder mehreren Frames (Interframe-Komprimierung) [3] unterschieden.

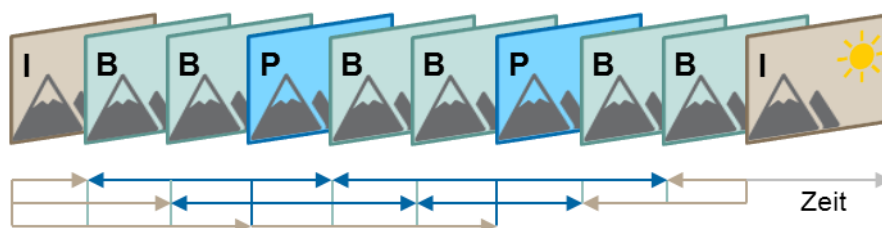


Abbildung 3: Komprimierung mit I-, B- und P-Frames

Die komprimierten Frames bei der Intraframe-Komprimierung werden als I-Frames bezeichnet und entsprechen im groben einem einzelnen JPEG. Bei der Interframe-Komprimierung werden nur die gleichbleibenden Bildelemente gespeichert, wobei zwischen Predicted-Frames (P-Frames) und Bidirectional-Frames (B-Frames) unterschieden wird. Die P-Frames werden ausschliesslich aus den vorhergehenden I-Frames errechnet und haben durchschnittlich die Hälfte der Grösse eines I-Frames. Die B-Frames werden auf den vorhergehenden und nachfolgenden I- und P-Frames errechnet und kommen auf etwa ein Viertel der Grösse eines I-Frames [4]. Die Pfeile in Abbildung 3 zeigen wie die I- und P-Frames in den anderen Frames verwendet werden.

Der H.264 hat sich wegen dessen hervorragenden Qualitäts-Komprimierungsverhältnissen [5] als einer der meist verwendeten Codecs durchgesetzt [6]. Weitere Beispiele zu Videocodecs wären VP8, MPEG-2 oder DivX.

### 2.1.3 Containerformat

---

Während sich ein Codec ausschliesslich mit dem Codieren von Daten befasst, so beschreiben Containerformate, wie die codierten Daten gespeichert werden. Diese Formate unterstützen meist mehrere Video- und Audiostreams, sowie auch Untertitel und weitere Metainformationen. Dies ermöglicht zum Beispiel in einer Datei Audiospuren für Deutsch und Englisch unterzubringen.

Beispiele zu Containerformaten wären AVI (Audio Video Interleave), MKV (Matroska Multimedia Container), WebM oder MP4 (MPEG-4 Part 14). Wobei das MP4 Containerformat zurzeit am weitesten verbreitet und verwendet wird [6].



## 2.2 Videostreaming

Das Übertragen von Videomaterial über das Internet ist mit der Zunahme von verfügbarer Übertragungsraten und Portalen wie YouTube in den letzten Jahren explosionsartig gestiegen. Doch während in den frühen Phasen ein Video im Internet nur betrachtet werden konnte, indem die komplette Datei heruntergeladen wurde, so entwickelten sich Schritt für Schritt neue Protokolle, welche kontinuierliches Betrachten und Herunterladen von Videoausschnitten ermöglichten. Diese Protokolle und Verfahren werden als Streaming bezeichnet.

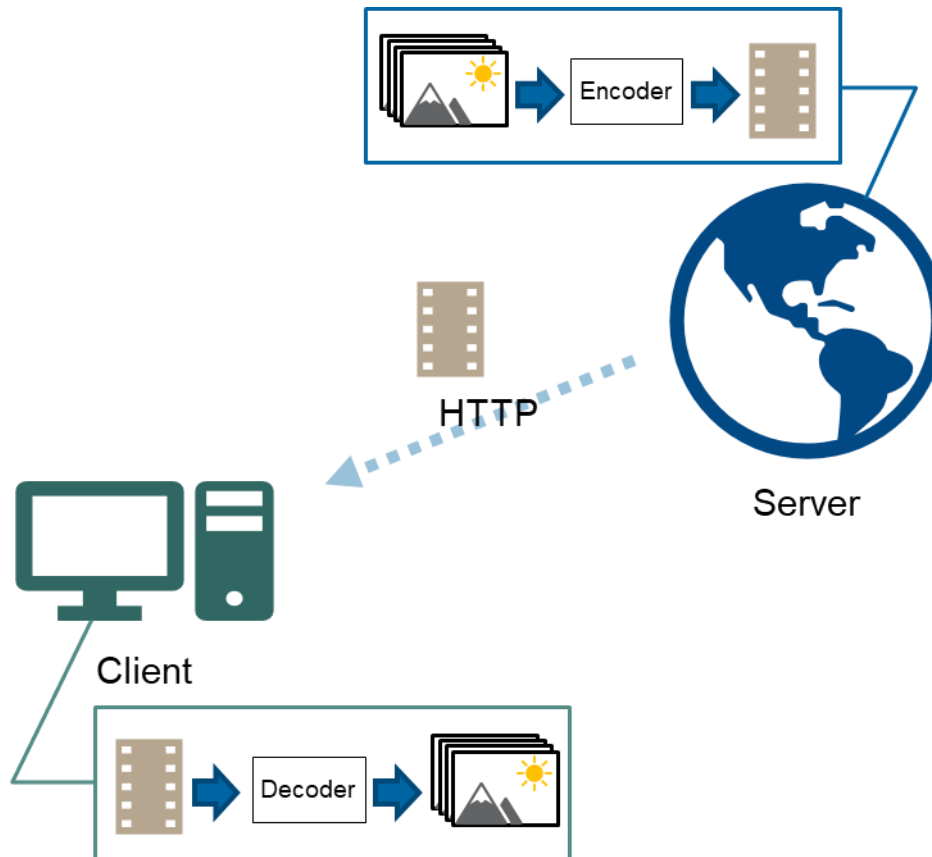


Abbildung 4: Übersicht Videostreaming

Die Videoinformationen werden auf dem Server codiert, meistens schon lange im Voraus, und dann über das Internet mit dem Hypertext Transfer Protocol (HTTP) an den Client verschickt, welcher das Video in Echtzeit wieder in einzelne Bilderabfolgen zerlegt und es dem Benutzer anzeigt (Abbildung 4).

### 2.2.1 Videostream und Segmente

Unter einem Videostream wird eine kontinuierliche Abfolge von Videomaterial verstanden, was es möglich macht, Videos über ein Netzwerk, wie zum Beispiel das Internet, zu schauen, ohne dass im Voraus das ganze Videomaterial heruntergeladen werden muss.

Dies wird insbesondere dadurch erreicht, dass die Informationen in einzelne Segmente aufgeteilt werden. Segmente können unabhängig von den vorigen Segmenten betrachtet, aber auch ohne Unterbruch hintereinander gereiht werden.

So lange das komplette Segment heruntergeladen wurde, kann der einige sekundenlange Videoausschnitt betrachtet werden, egal wie stabil die Internetverbindung in diesem Moment ist. Bleibt die Verbindung stabil, sollte es möglich sein das nächste Segment herunterzuladen, bevor das aktuelle Segment fertig abgespielt wurde.

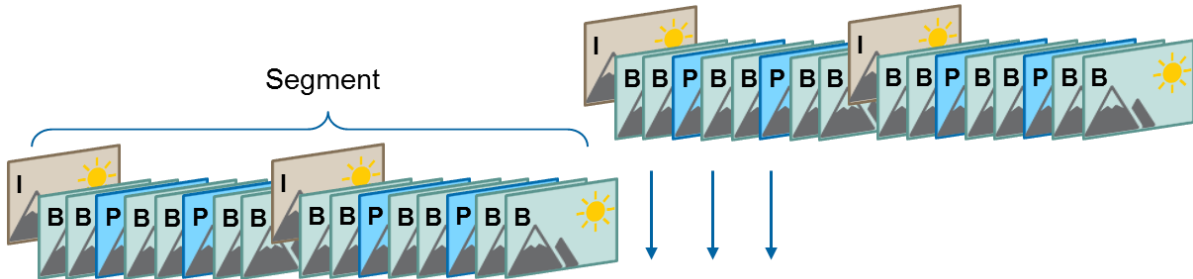


Abbildung 5: Grober Aufbau eines Segments

Wie in Abschnitt 2.1.2 erklärt, wird ein Video in I-, B- und P-Frames codiert, wobei ein I-Frame jeweils ein gesamtes Videoframe enthält. Ein Segment endet immer genau vor dem nächsten I-Frame [7], womit das nächste Segment mit einem I-Frame beginnt und somit jedes Segment unabhängig von vorigen Segmenten ist. Diese Aufteilung kann auch in Abbildung 5 erkannt werden.

Eines der meist angetroffenen Segmentformate ist der MPEG Transport Stream (MPEG-TS). Das Format wird nicht nur für Streaming im Internet verwendet, sondern hält auch Einzug bei verschiedensten Übertragungssystemen wie zum Beispiel Digital Video Broadcasting (DVB) [8]. Ein weiterer Vorteil von der Segmentierung eines Videostreams ist, dass die Segmente beliebige Auflösungen oder Datenraten haben können und ein Videostream somit dynamisch angepasst werden kann. Das heisst, wenn die Verbindungsqualität plötzlich einbricht und das Laden des nächsten Segments länger dauert als das Abspielen des aktuellen Segments, so kann ein intelligenter Videoplayer automatisch das Segment mit geringerer Videoqualität und deshalb auch geringere Segmentgrösse anfordern.

## 2.2.2 Buffering

Beim Buffering von Videostreams werden zukünftige Segmente im Voraus heruntergeladen und lokal im Speicher gehalten. Falls die Internetverbindung sich über eine kurze Zeit plötzlich verschlechtern würde, kann das Video ohne Unterbrechung weiter abgespielt werden, da eine gewisse Anzahl von Sekunden bis Minuten an Videomaterial bereits lokal vorhanden ist.

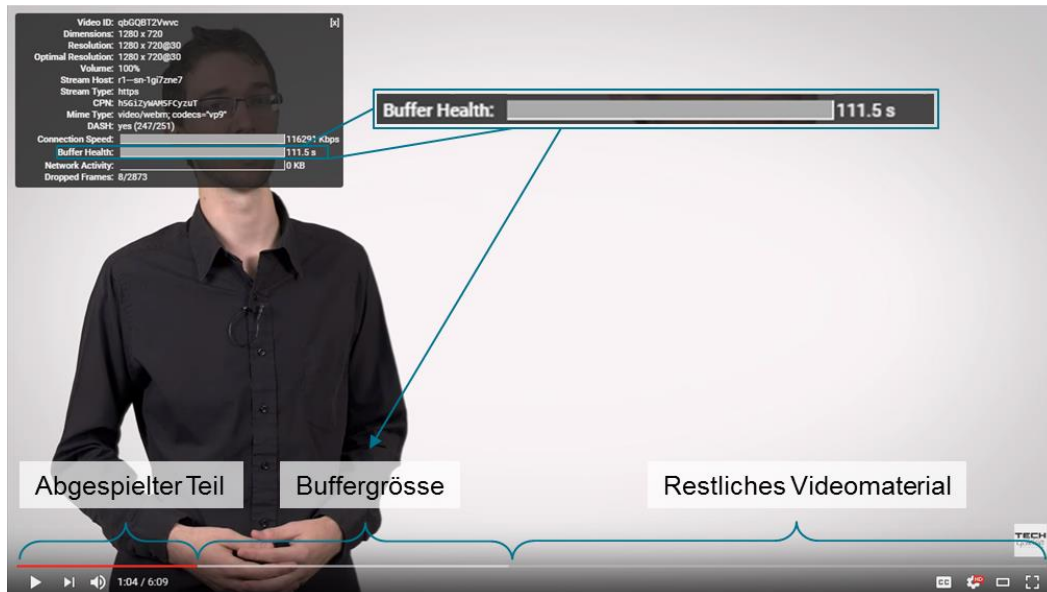


Abbildung 6: Puffergrösse im YouTube Player

Das Füllen des Buffers mit dem gesamten Video ist nicht immer intelligent, denn nicht jede Person, welche einen Videostream öffnet, möchte das gesamte Video betrachten. Somit wäre das vollständige Buffern eines Videos in diesen vielen Fällen eine Verschwendung von übertragenen Daten, was sich insbesondere bei beschränktem Datenvolumen bemerkbar machen würde. Der YouTube Player verhält sich beim Buffering sehr intelligent und lädt nicht immer das gesamte Video, wie auch in Abbildung 6 zu sehen ist.

### 2.2.3 Replay und Live Videos

Es wird unterschieden zwischen Echtzeit- und zeitunabhängigen Videostreams. Dabei ist es wichtig die zwei Begriffe Live Videos und Replay Videos zu verstehen. Live Videos sind Videos, welche sehr zeitnah, also nur mit einer kurzen Verzögerung von einigen Sekunden bis maximal wenigen Minuten, gestreamt werden. Replay Videos sind Videos, für welche genügend Zeit zur Verfügung steht, um verschiedenste Post-Processing Schritte durchzuführen. Live Videostreams findet man insbesondere bei Fernsehübertragungen, während sich zum Beispiel YouTube zum grössten Teil nur mit Replay Videostreams befasst.

### 2.2.4 Unterbrechungen

#### 2.2.4.1 Überspringende Unterbrechung

Bei der überspringenden Unterbrechung werden Teile des Videostreams dem Benutzer nicht gezeigt, sondern der Stream fährt nach dem Unterbruch an einer anderen Stelle weiter als zuvor. Diese Art von Unterbrechung ist insbesondere bei Live Videos der Fall, da diese in Echtzeit ablaufen und man mit jedem Warten immer weiter nach hinten im Stream fallen würde. Dies wäre nicht immer im Interesse des Betrachters, insbesondere wenn dieser sich zum Beispiel einen Fussballmatch ansehen möchte und nach einer Weile merkt, dass der Videostream um einige Minuten zum eigentlichen Match verschoben ist.

## 2.2.4.2 Stockende Unterbrechung

Eine stockende Unterbrechung tritt meist bei Replay Videos auf, wenn die Übertragungsgeschwindigkeit zu klein ist und das Buffering nicht ausreicht. Dabei kommt es zum Anhalten der Wiedergabe und erst nach einer bestimmten Zeit wird der Stream an derselben Stelle weitergeführt. Viele Videoplayer zeigen in diesem Fall an, wie lange es geht bis der Buffer wieder gefüllt wurde und die Wiedergabe weitergeführt werden kann.

## 2.2.5 Transport Protokoll

---

### 2.2.5.1 Transmission Control Protocol (TCP)

TCP ist ein zuverlässiges, geordnetes und fehlerbehebendes Protokoll, welches sich im Webbereich und vielen anderen Applikationen als das vorherrschende Protokoll durchgesetzt hat. Da HTTP eine zuverlässige und geordnete Verbindung voraussetzt [9], bietet sich TCP als optimales Protokoll für das Stream von Videos über HTTP an. Zusätzlich sind die beiden Ports 80 und 443 für normalen HTTP und TLS verschlüsselten HTTP Verkehr bei fast allen Firewalls geöffnet und somit steht dem Übertragen von Videos auf diesem Weg wenig im Wege.

### 2.2.5.2 User Datagram Protokoll (UDP)

Als Gegenstück zu TCP, gibt UDP keine Garantie für die Zuverlässigkeit und die Datenpakete können in beliebiger Reihenfolge ankommen. Das Protokoll gewinnt aber dadurch eine höhere Durchsatzrate, insbesondere auch bei schlechteren oder verstopften Verbindungen.

Bei den untersuchten Videoportalen wird UDP nicht verwendet, sondern es findet mehr Einsatz bei Videostreaming über Set-Top-Boxen.

### 2.2.5.3 Quick UDP Internet Connections (QUIC)

Bei viel Traffic auf einer Leitung oder bei schlechter Verbindung kann TCP schnell zu einem Problem werden, da die einzelnen Datenpakete immer wiederholt werden und sich der Durchsatz immer mehr verschlechtert. Google hat mit dem QUIC Protokoll ein alternatives Transportprotokoll entwickelt, welches auf UDP basiert und Zuverlässigkeit so wie richtige Ordnung der Pakete im QUIC Protokoll definiert. Durch die Unabhängigkeit von TCP kann QUIC einen besseren Durchsatz erlangen, wird aber Grossteils nur von Google Diensten unterstützt und kommt nur im Google Chrome zum Einsatz.

## 2.2.6 Streaming Protokoll

---

### 2.2.6.1 HTTP Live Streaming (HLS)

HLS wurde von Apple Inc. entwickelt [10] und wird deshalb auch von QuickTime und Safari unterstützt. Das Protokoll basiert auf dem HTTP Protokoll und ist somit optimal geeignet für das Streamen über das Internet, da HTTP Traffic praktisch durch jede Firewall kommt. Das Protokoll erzwingt die Verwendung des H.264 Videocodec, kann jedoch mit AAC, MP3, AC-3 oder EC-3 als Audiocodec arbeiten. Der Encoder verpackt die Video- und Audioinformationen in einen MPEG-2 Transport Stream. Diese Segmente sind in der Regel um die fünf Sekunden lang. Zusätzlich zu den einzelnen Segmenten, wird eine Indexdatei erstellt, welche dem Client die verschiedenen Qualitäten des Streams bekannt gibt.

## 2.2.6.2 Dynamic Adaptive Streaming over HTTP (DASH)

DASH wurde durch die Moving Picture Experts Group (MPEG) erstellt und international standardisiert [11]. Technisch gibt es einige Gemeinsamkeiten mit HLS. Wie HLS verwendet DASH ebenfalls HTTP, des Weiteren werden Videostreams auch in kleinere Segmente aufgeteilt. Diese sind mit durchschnittlich drei Sekunden ein wenig kürzer als bei HLS. Zusätzlich zu den einzelnen Segmenten wird eine Indexdatei erstellt, welche dem Client die verschiedenen Qualitäten des Streams bekannt gibt. Ein grosser Vorteil von DASH ist, dass das Protokoll nicht nur den H.264 Videocodec unterstützt, sondern auch Codecs wie VP9 und H.265.

## 2.3 Apposite Linktropy 5500

Der Apposite Linktropy 5500 (Abbildung 7) ist ein WAN (Wide Area Network) Emulator, mit dem Netzwerkverbindungen unter bestimmten, realen Bedingungen getestet werden können.

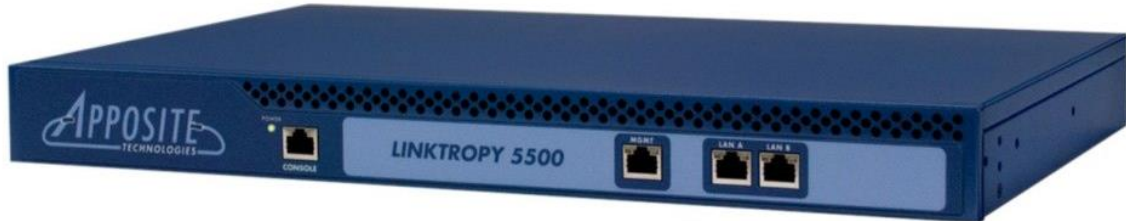


Abbildung 7: Apposite Linktropy 5500

Dabei simuliert der Apposite Linktropy eine einzelne Verbindung über zwei 1 Gbit/s Schnittstellen und setzt sich somit als Mittelstück in die Netzwerkkommunikation zweier Geräte. Da eine durchgehende Verbindung simuliert wird, merkt der Client nichts davon, denn der Traffic zum Server im Internet geht wie gewohnt vonstatten. Der Apposite Linktropy wird mithilfe einer intuitiven, graphischen Benutzeroberfläche durch den Controller gemanagt. Das Setup in Abbildung 8 ist schnell aufgebaut und innerhalb weniger Minuten kann mit dem Testen verschiedener Netzwerkszenarien begonnen werden. Daher eignet er sich perfekt für die Streaming- und Videoportalverhaltensanalyse dieser Arbeit.

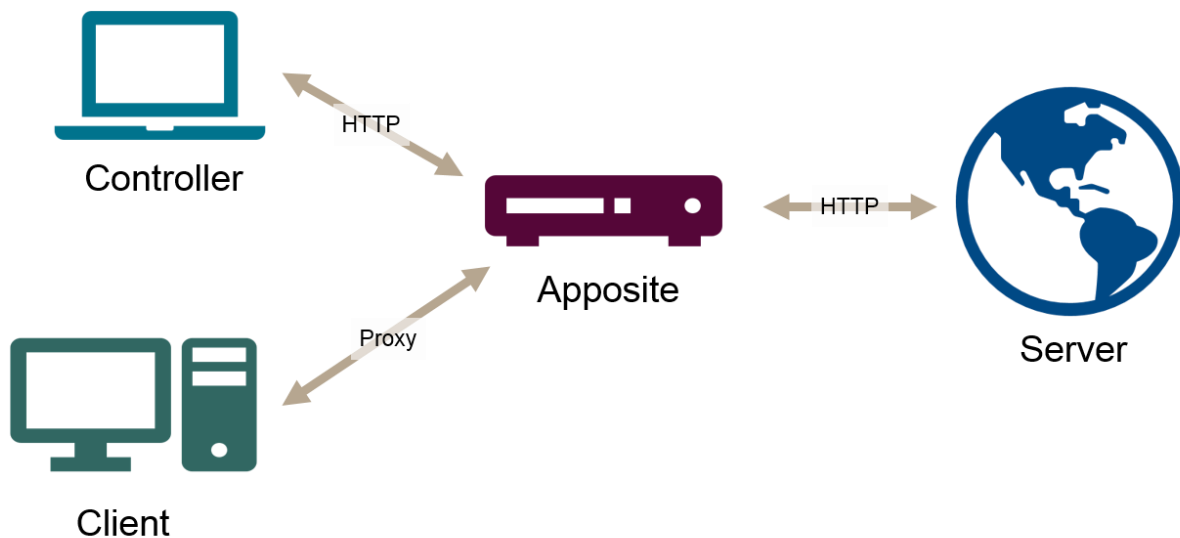
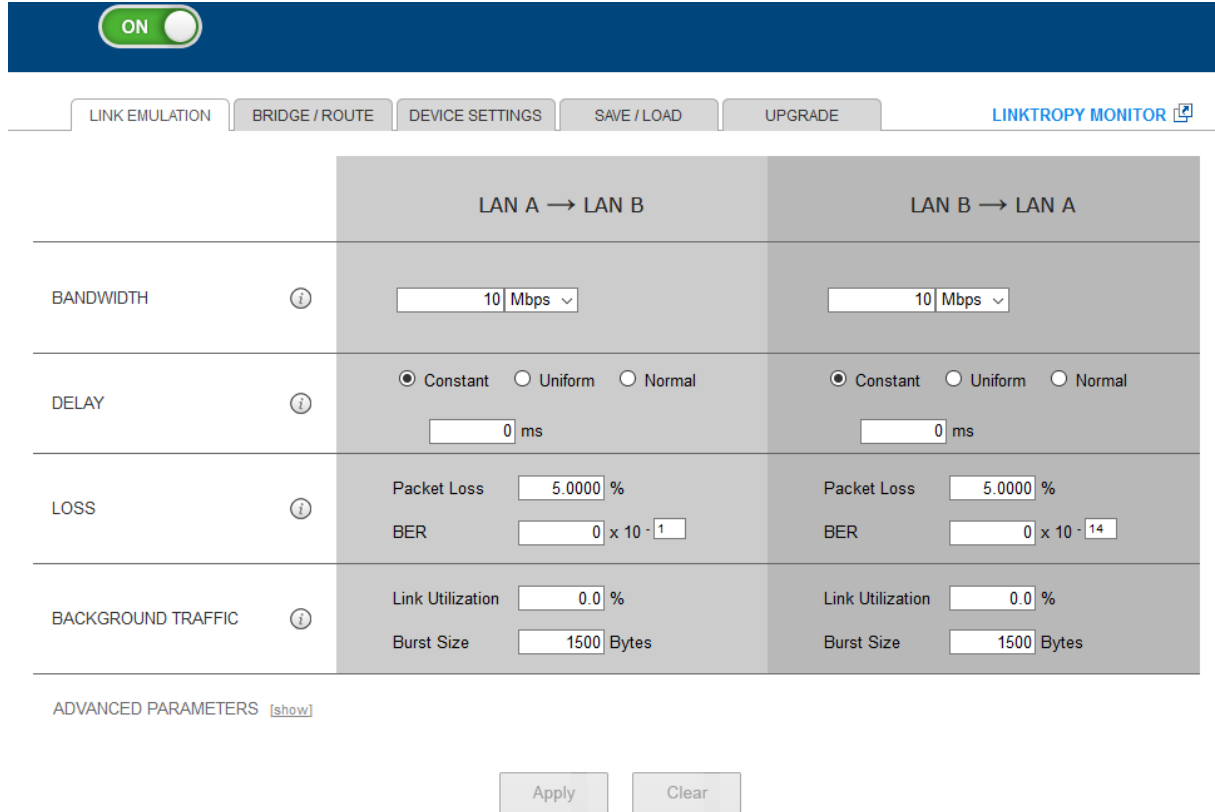


Abbildung 8: Setup mit Apposite Linktropy

### 2.3.1 Einstellungen

Der Apposite Linktropy 5500 hat eine Vielzahl von verfügbaren Optionen und Einstellungen. Es werden nicht alle vorhandenen Optionen erklärt, sondern nur eine Auswahl an Einstellungsmöglichkeiten, welche im Verlauf dieses Projekts sinnvoll waren und benutzt wurden.



The screenshot shows the configuration interface for Linktropy 5500. At the top, there is a green 'ON' toggle switch. Below it are navigation tabs: 'LINK EMULATION', 'BRIDGE / ROUTE', 'DEVICE SETTINGS', 'SAVE / LOAD', and 'UPGRADE'. A 'LINKTROPY MONITOR' link is also present. The main configuration area is divided into two columns: 'LAN A → LAN B' and 'LAN B → LAN A'. The settings are as follows:

	LAN A → LAN B	LAN B → LAN A
<b>BANDWIDTH</b>	10 Mbps	10 Mbps
<b>DELAY</b>	Constant (selected), Uniform, Normal 0 ms	Constant (selected), Uniform, Normal 0 ms
<b>LOSS</b>	Packet Loss: 5.0000 % BER: 0 x 10 <sup>-1</sup>	Packet Loss: 5.0000 % BER: 0 x 10 <sup>-14</sup>
<b>BACKGROUND TRAFFIC</b>	Link Utilization: 0.0 % Burst Size: 1500 Bytes	Link Utilization: 0.0 % Burst Size: 1500 Bytes

At the bottom, there is an 'ADVANCED PARAMETERS [show]' link and two buttons: 'Apply' and 'Clear'.

Abbildung 9: Apposite Einstiegsseite

Während der Analyse wurden die folgenden drei Einstellungen vielfach verwendet:

#### Bandwidth (Bandbreite)

Hiermit kann die Datenübertragungsrate auf eine beliebige Geschwindigkeit eingestellt werden. Zur Auswahl steht eine enorme Spannweite von unglaublich schleppenden 300 bit/s bis zu schnellem 1 Gbit/s. Für die Analysen wurde eine bestimmte Anzahl verschiedener Datenübertragungsraten zwischen 1 Mbit/s bis zu 1 Gbit/s getestet.

## Loss (Paketverlust)

Diese Einstellung definiert wie viele Pakete vom Apposite Linktropy weggeworfen werden. Mit einem 5% Paketverlust, wie im Einstiegseitenbeispiel angegeben, verwirft der Apposite Linktropy jedes zwanzigste Paket. Das bedeutet, dass jedes zwanzigste Paket vom Sender beim Empfänger und umgekehrt ohne Begründung nicht ankommt. Getestet wurde anfänglich ein Paketverlust von 5% und später hauptsächlich ein Paketverlust von 10%, wobei zum Teil ein Paketverlust von bis zu 20% definiert wurde. Das bedeutet, dass jedes vierte Paket den Empfänger nicht erreicht. Diese Einstellung macht sich, je höher desto schneller, besonders beim TCP Handshake bemerkbar, da Stream-Aufbauten relativ schnell ungültig sein können.

## Background traffic (Hintergrundverkehr)

Soll auf der Leitung Verkehr im Hintergrund simuliert werden, ist diese Einstellung dafür vorgesehen. Zum einen kann eingestellt werden wie oft die Leitung besetzt wird (Link Utilization) und zum anderen wie gross die einzelnen Pakete sein sollen (Burst Size). Die Burst Size wurde nicht verändert, sondern auf den typischen 1500 Bytes gemäss der Maximum Transmission Unit für Ethernet belassen. Die Verwendung im Hintergrund wurde normalerweise auf 25% gesetzt, was bedeutet, dass das Netzwerk zu 25% der Zeit mit anderen Paketen gefüllt ist und unserem Stream nur 75% der Leitung zur Verfügung stehen.

## 2.3.2 Scheduler

The screenshot shows the 'Linktropy Monitor - Mozilla Firefox' browser window displaying the configuration page for a traffic profile. The profile is named 'LAN A → LAN B' and 'LAN B → LAN A'. The configuration is divided into several stages, each with specific parameters for Rate, Delay, Loss, BER, Duplication, Queue, Background Traffic, and Reordering. The 'Duration' for each stage is set to 30 minutes. The 'Intermediate Steps' and 'Step Duration' are also configurable.

Stage	Direction	Rate	Delay	Loss	BER	Duplication	Queue	Background Traffic	Reordering
Stage 1	LAN A → LAN B	1000 Mbps	0 ms	0%	0	0%	250 ms	0%	0%
Stage 2	LAN A → LAN B	10 Mbps	0 ms	0%	0	0%	250 ms	0%	0%
Stage 3	LAN A → LAN B	5 Mbps	0 ms	0%	0	0%	250 ms	0%	0%
Stage 4	LAN A → LAN B	2.5 Mbps	0 ms	0%	0	0%	250 ms	0%	0%
Stage 5	LAN A → LAN B	2 Mbps	0 ms	0%	-	-	-	-	-

Abbildung 10: Der Apposite Linktropy Scheduler



Eine für diese Arbeit wichtige Option ist der eingebaute Scheduler wie in Abbildung 10 dargestellt. Mit diesem können einzelne Szenarienschritte einfach und bequem simuliert und getestet werden. Zum Beispiel wurde eine Bahnverbindung simuliert, in welcher die Verbindung zeitweise gut läuft und dann für einige Minuten eine geringere Datenübertragungsrate erfährt oder die Verbindung ganz abbricht und der Paketverlust auf 100% springt.

Um später aussagekräftige Auswertungen zu machen ist eine Quantität der Daten ausschlaggebend. Damit die erfassten Parameterdaten nicht immer gleich sind, wurde ein Scheduler wie im Bild ersichtlich konfiguriert. Dabei gibt es zehn verschiedene Stufen, jede 30 Minuten lang, die in einer Schleife nacheinander durchgegangen und simuliert werden. Angefangen mit einer sehr guten Verbindung fällt die Datenübertragungsrate in mehreren Schritten auf 2 Mbit/s. Anschliessend gesellt sich ein 10% Paketverlust dazu. Dieser verbleibt die restlichen 5 Stufen wobei die Datenübertragungsrate wieder bis zu 1 Gbit/s ansteigt. Danach fängt das ganze Szenario wieder von vorne an. Dadurch konnte eine viel unterschiedlichere Datensammlung erfolgen, was schlussendlich zu einer besseren Auswertemöglichkeit führte.

## 2.4 OWASP Zed Attack Proxy (ZAP)

Der ZAP Proxy ist das aktivste Projekt [12] des Open Web Application Security Project (OWASP). Der Proxy unterstützt neben dem Abfangen von Proxy-Traffic auch passives Scannen, einen AJAX Web Crawler, Fuzzer, WebSocket Support und einiges mehr.

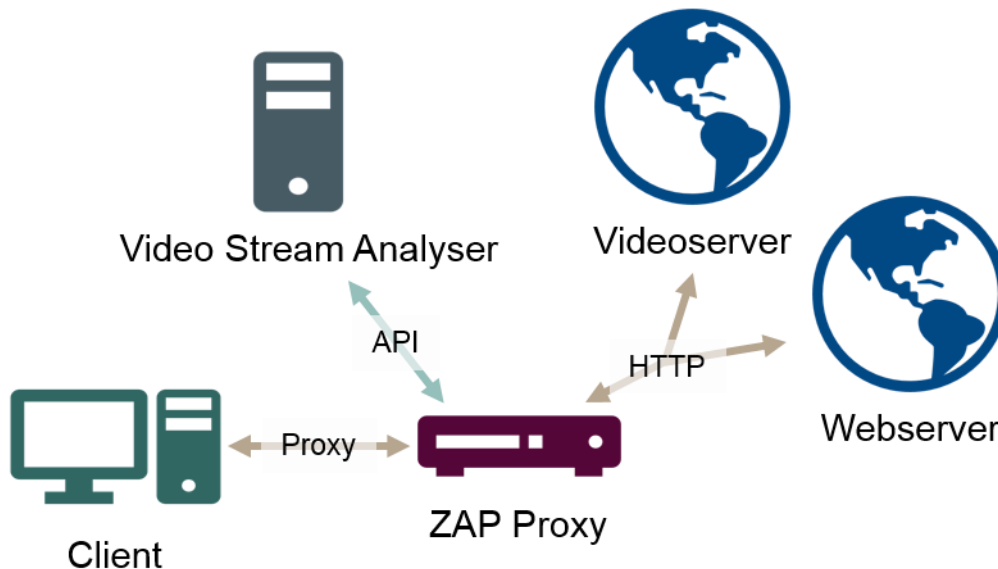


Abbildung 11: Verwendung von ZAP

In der Arbeit wird der ZAP Proxy ausschliesslich für das Abfangen von Internet-Traffic verwendet. Die abgefangenen Informationen werden dann über das Application Programming Interface (API) auf dem Video Stream Analyser verarbeitet.

### 2.4.1 HTTPS Verkehr

Gewisse Videostreams und Webseiten werden ausschliesslich verschlüsselt über HTTPS verschickt. Dies macht es dem Proxy unmöglich die gesendeten Daten abzufangen. Doch mit der Installation des vom ZAP Proxy ausgestellten Root-Zertifikat, wird dem Browser eine sichere Verbindung zum Proxy vorgespielt und der Proxy kann die abgefangenen Informationen entschlüsseln bevor sie von einem Server an den Client weitergeleitet werden.

### 2.4.2 Application Programming Interface (API)

Der ZAP bietet direkt eine API an, welche auf verschiedene Wege verwendet werden kann. Der schnellste und einfachste Zugang erhält man über den Webbrowser als eine Art REST API. Für Python 2.7 existiert jedoch auch ein Paket, welches die API als Python Funktionen anbietet.

## 2.5 Splunk

Splunk Inc. [13] ist eine Firma die ihre Software unter dem Namen Splunk vertreibt. Dabei gibt es zwei Produktvarianten, zum einen die Splunk Enterprise und zum anderen die Splunk Cloud. Die Erklärung hier bezieht sich auf Splunk Enterprise, wobei die Splunk Cloud im Allgemeinen dasselbe ist wie die Splunk Enterprise, mit dem wichtigen Unterschied, dass die Daten in der Cloud gespeichert werden und nicht mehr auf lokalen Geschäftsservern. Splunk Enterprise kann für einen Monat getestet werden, danach muss entweder eine Lizenz gekauft werden oder es wird auf eine freie Version (Splunk Light) umgeschaltet. Splunk kommt hauptsächlich zur Anwendung bei der Logdatenverarbeitung. Diese kann in drei aufeinanderfolgende Punkte unterteilt werden.

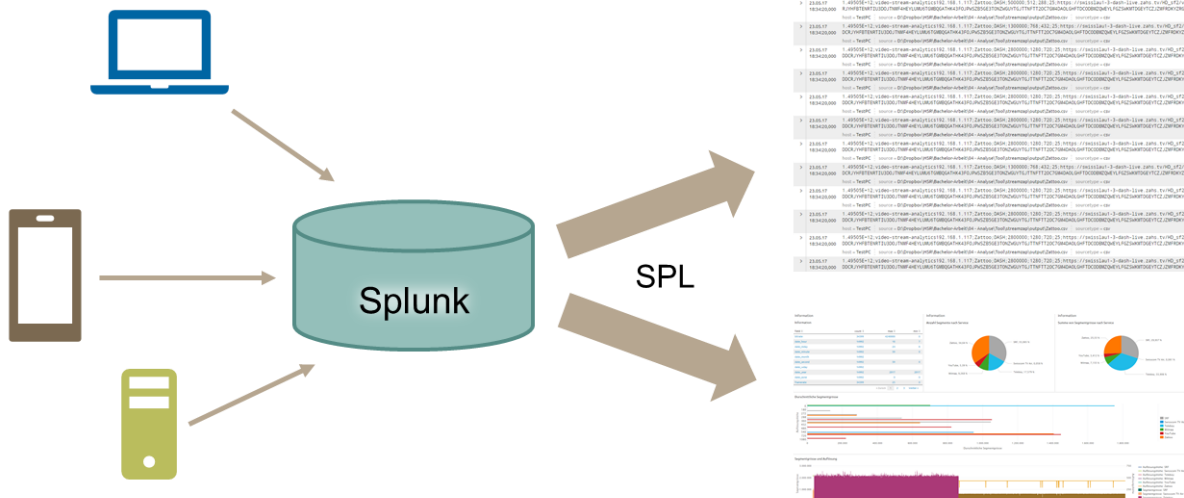


Abbildung 12: Funktionsweise von Splunk

### 2.5.1 Erfassung

Es können fast beliebige maschinelle Daten aus praktisch jeder Datenquelle und jedem Speicherort in Echtzeit gesammelt und indiziert werden. Man kann sich Splunk als einen offenen Behälter vorstellen, in den man alle gewünschten Daten hineinwerfen kann. Das wichtigste ist, dass jedes Ereignis mit einem Zeitstempel versehen ist. Falls kein Zeitstempel mitgegeben wird, so vergibt Splunk den Daten einen Zeitstempel mit der Zeit als das Ereignis an Splunk gesendet wurde.

Die Daten können ohne Schema erfasst werden und Splunk wird automatisch die Zuordnung finden. Für das bessere Filtern und einfachere Auswerten empfiehlt es sich jedoch Splunk eine Art Schema mitzuteilen.

### 2.5.2 Auswertung

Die Auswertung findet mit der Splunk-eigenen Suchsprache SPL (Search Processing Language) statt. Zahlreiche Funktionen, wie zum Beispiel Echtzeitsuchanfragen, Mustererkennung und vieles mehr, erlauben ein umfangreiches Auswerten der Daten.

Es ist empfehlenswert für jede Kategorie von Ereignissen einen eigenen Index zu erstellen, da die Suche darauf aufbaut und die Daten nicht durchmischt werden.

## 2.5.3 Visualisierung

---

Sind die Daten ausgewertet, können sie anschliessend visualisiert und mit verschiedenen Graphen und Diagrammen dargestellt werden. Dabei können diese Visualisierungen in Dashboards zusammengefasst und übersichtlich dargestellt werden. Dashboards bieten zusätzlich noch weitere Konfigurationsmöglichkeiten (z.B. Achsenbeschriftungen, bestimmte Farbgebung, vordefinierte Diagrammtypen) in dem die Darstellung per Extensible Markup Language (XML) noch erweitert werden kann.

### 3 Analyse von Videoportalen

#### 3.1 Einführung

##### 3.1.1 Ziel

Ziel dieser Analyse ist es, messbare Merkmale zum Kundenerlebnis zu definieren, Daten dazu zu erfassen und Unterschiede bei verschiedenen Videoportalen zu erkennen.

##### 3.1.2 Vorgehen

Es werden Messungen zu verschiedensten Videoportalen im Internet mit unterschiedlichen Datenübertragungsraten, Paketverlust und Background Traffic Werten durchgeführt und deren Auswirkung auf das Kundenerlebnis bestimmt. Es ist auch der Einfluss von Werbung auf das Kundenerlebnis zu betrachten. Die grundsätzlichen Funktionsweisen der verschiedenen Videoportale werden studiert und verglichen.

#### 3.2 Videoportale

##### 3.2.1 USA

Die USA ist, wie in vielen anderen Bereichen, zurzeit einer der grössten Spieler im Streaming Markt.

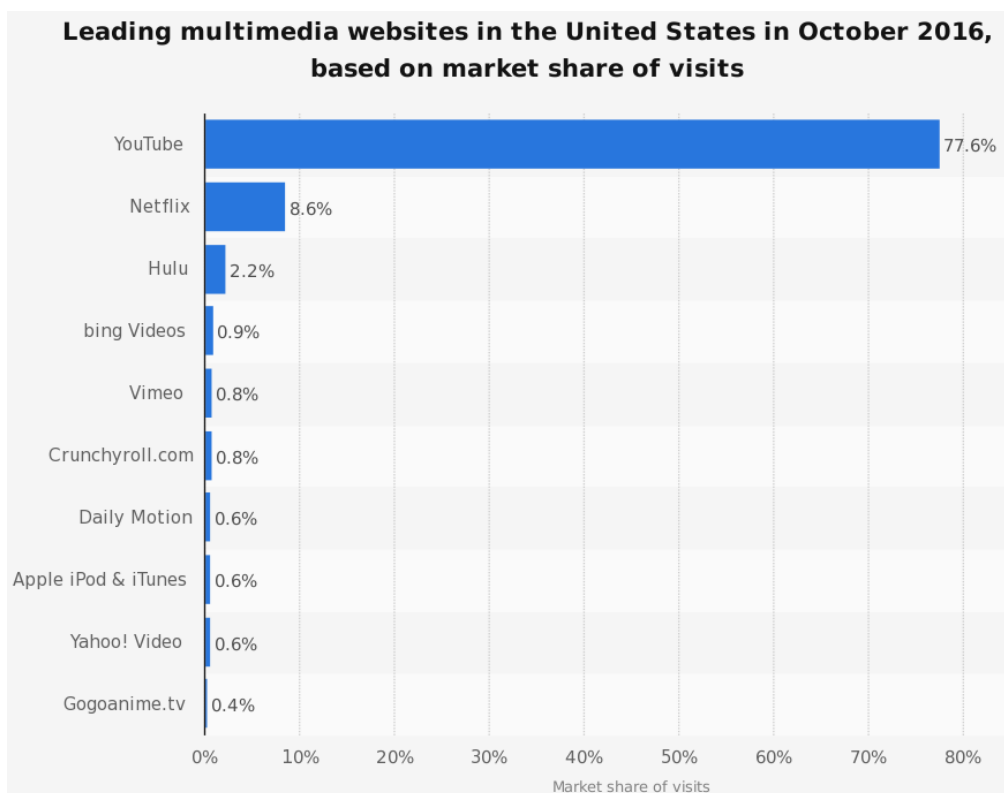


Abbildung 13: Führende Multimedia Webseiten in den USA (2016)

Aus Abbildung 13 ist auf den ersten Blick ersichtlich, dass YouTube in den USA den grössten Marktanteil aller Webseitenaufrufe hat und dies mit riesigem Abstand zum zweitplatzierten Netflix. Über 77% aller Aufrufe im Oktober letzten Jahres waren an YouTube gerichtet, während Netflix auf 8.6% Marktanteil kam. Wobei sich die Prozentangaben auf die Anzahl Besucher der einzelnen Seiten verteilt. Etwas erstaunlich ist die Aufruf-Differenz zwischen YouTube und Netflix schon, denn über die Hälfte aller Haushalte in den USA haben einen Netflix-Zugang [14]. Wenn man jedoch bedenkt, dass sich das Angebot von Netflix auf Filme und Fernsehsendungen beschränkt, während YouTube von Unterhaltung aller Art, über vollumfängliche Filme, bis hin zu Musik fast alles zu bieten hat.

### 3.2.2 Deutschland

Diese Untersuchung wurde in Deutschland durchgeführt, da die Erhebungen für die Schweiz noch geringer und oberflächlicher sind als in Europa allgemein. In Europa sind solche Statistiken, wie aus den USA, schwieriger zu finden, da die meisten Erhebungen und Umfragen alle Portale zusammennehmen. Das heisst fast alle Statistiken geben die Verbreitung von Videoportalen wieder, aber nicht wie gross der Anteil der einzelnen Portale an der gesamten Benutzung ist. Die Annahme war, dass Deutschland als grosser Spieler in Europa bessere Abbildungsmöglichkeiten auf die Schweiz bieten würde. In einer Online Umfrage von Goldmedia GmbH [15] mit etwas mehr als 2000 Befragten ergab sich für Deutschland folgendes Bild.

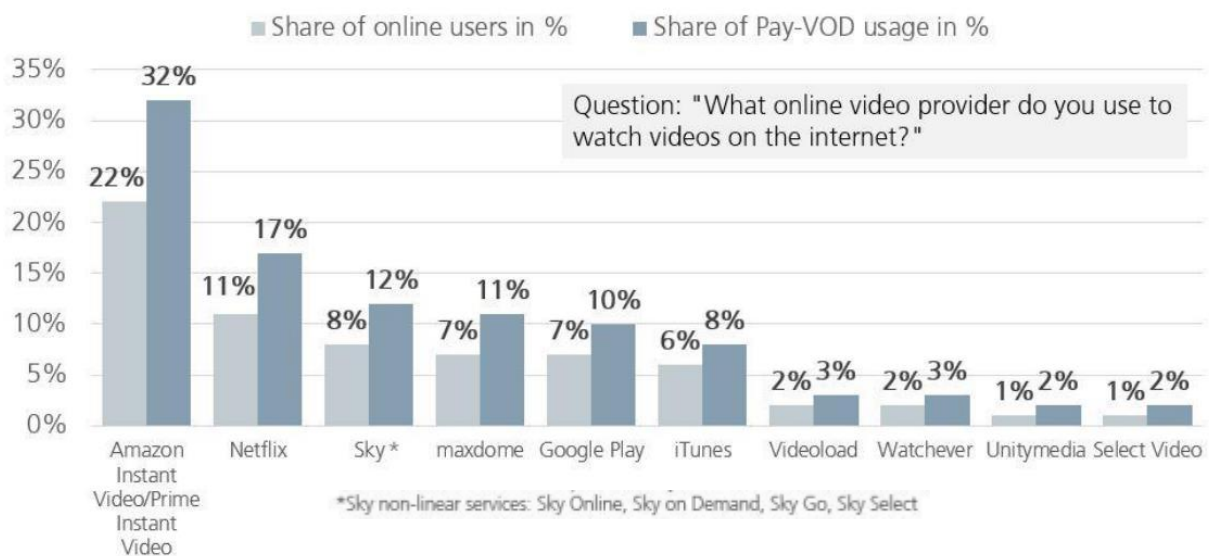


Abbildung 14: Videoportale in Deutschland (2016)

Da die Statistik in Abbildung 14 nur kostenpflichtige Video-on-Demand (VoD) Angebote enthält, kommt YouTube nicht vor. Man darf davon ausgehen, dass YouTube in Europa wie in den USA, den ersten Platz einnehmen würde.

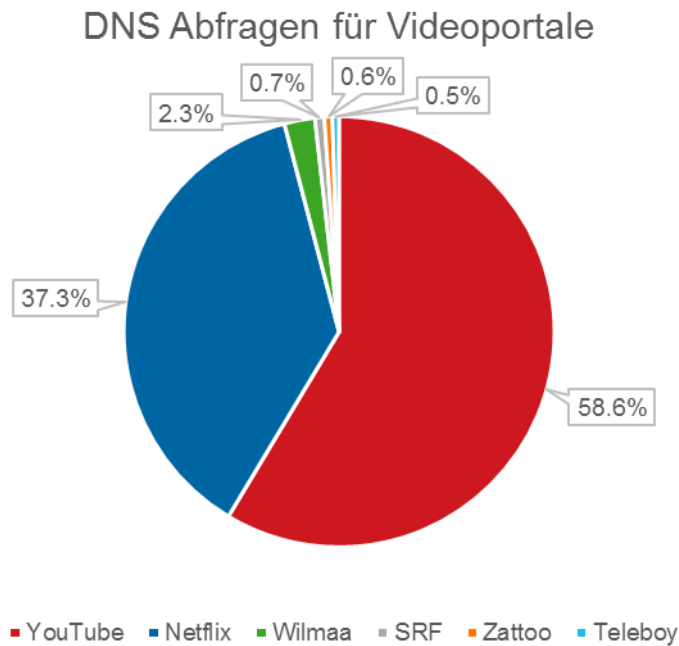
Bei den kostenpflichtigen Angeboten hat Amazon Prime die grösste Beliebtheit erhalten, wobei erkennbar ist, dass 22% aller befragten Personen dieses Angebot nutzen, was ein Marktanteil von 32% aller bezahlten VoD-Angebote ausmacht.

Die Analyse ergab, dass eine Abbildung auf die Schweiz schwierig und nicht nutzbringend ist, da einige der in Deutschland angebotenen Portale in der Schweiz nicht zur Verfügung stehen.

#### 3.2.3 Schweiz

Es gibt keine öffentlich zugänglichen Nutzungsstatistiken über Schweizer Videoportale, weshalb stattdessen Statistiken zu Webseitenaufrufen betrachtet werden.

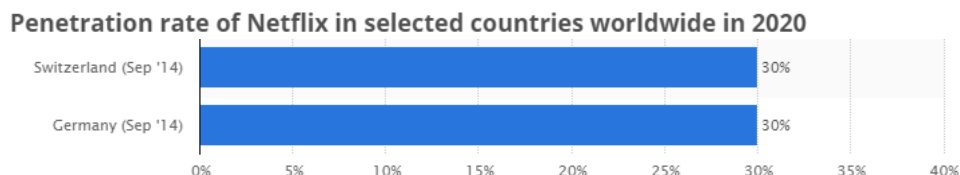
Aussagekräftige Daten über gemessene DNS Aufrufe bei Swisscom für verschiedene Schweizer Videoportale wurden von cnlab zur Verfügung gestellt [16].



**Abbildung 15: DNS Abfragen für Videoportale bei Swisscom via cnlab**

Für das Verständnis der Abbildung 15 ist wichtig zu wissen, dass sich die Prozentangaben ausschliesslich auf die sechs aufgeführten Portale beziehen, sich dazwischen jedoch keine weiteren Videoportale befinden.

Hier zeichnet sich mit YouTube ein ähnliches Bild ab wie in den USA, denn auch in der Schweiz ist YouTube mit 58.6% aller DNS Abfragen für Videoportale der unangefochtene Spitzenreiter. An zweiter Stelle steht Netflix mit 37.3%. Das ist insofern interessant, da der momentane Verbreitungsgrad von Netflix in der Schweiz noch in der Startphase ist. Aktuelle Prognosen gehen von einer 30% Haushaltsdurchdringung im Jahr 2020 aus [17].



**Abbildung 16: Prognose zum Verbreitungsgrad von Netflix in der Schweiz und in Deutschland**

Die Angaben in Klammern beziehen sich auf den Netflixstart im jeweiligen Land.

### 3.2.4 Providers

Mit Hilfe von den Statistiken und der Bekanntheitsgrade von Videoportalen bei den Studenten und Betreuern, wurden weitere Untersuchungen auf die Liste in Abbildung 17 beschränkt:



Abbildung 17: Ausgewählte Videoportale

Die Auswahl setzt sich insbesondere aus populären Providern mit einer Vielzahl von Fernsehkanälen zusammen. Ausserdem werden nur die Gratisangebote betrachtet, mit Ausnahme von Netflix wegen dessen grosser Popularität.



### 3.3 Technischer Aufbau

#### 3.3.1 Aufbau

Der Experimentaufbau kann im groben aus Abbildung 18 entnommen werden.

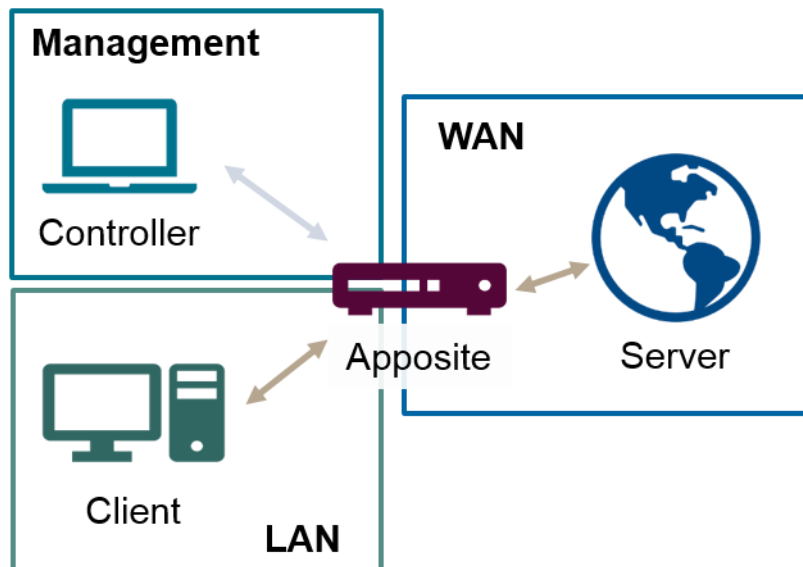


Abbildung 18: Experiment Aufbau

Das Herzstück des Aufbaus ist der Apposite Linktropy 5500 (siehe Abschnitt 2.3), welcher es ermöglicht verschiedenste WAN Situationen, wie zum Beispiel Übertragungsrateneinschränkungen oder Paketverluste zu simulieren und dient auch zur Messung der aktuellen Übertragungsrate.

Der Labor Computer (Client) wird über den Apposite mit dem Internet verbunden, über das die Server der untersuchten Videoportale erreicht werden können.

Da das Controller Notebook über einen separaten Netzwerkport angeschlossen wird, kann eine direkte Verbindung zum Apposite hergestellt werden um die Einstellungen der Simulation zu verändern.

#### 3.3.2 Vorbereitung

Durch Kontrolle der laufenden Prozesse im Task-Manager und Ressourcenmonitor sowie der Betrachtung der Datenübertragungsnutzung in Apposite Linktropy 5500 wurde für die Datenerfassung auf dem Labor Computer überprüft, dass keine weiteren Dienste im Hintergrund das Netzwerk belasten. Somit konnte sichergestellt werden, dass die Resultate frei von kontrollierbaren Fremdeinflüssen sind und dadurch nicht beeinträchtigt wurden.

### 3.3.3 Erfassung

---

#### 3.3.3.1 Auflösung

Mit der grössten Übertragungsrategieinstellung von 1 Gbit/s wurde der Videostream gestartet und dann in den Stufen 5 Mbit/s, 2.5 Mbit/s, 2 Mbit/s, 1.5 Mbit/s, 1 Mbit/s und 500 Kbit/s angepasst, während die Videoqualitätseinstellung mit den Informationen aus den einzelnen Videoplayer oder über den Netzwerk Tab in den Chrome Entwicklungstools erfasst wurden.

#### 3.3.3.2 Startverzögerung

Für die manuelle Messung der Startverzögerung wurden die einzelnen Streams bei jeder neu eingestellten Konfiguration (z.B. Übertragungsrate) durch Aktualisierung des Videoportals neu gestartet und anschliessend die Zeit erfasst.

#### 3.3.3.3 Zeitmessung

Für Zeitmessungen im Sekundenbereich wie zum Beispiel für die Startverzögerung oder Recoverytime wurde die Stoppuhrfunktion des persönlichen Smartphones verwendet. Durch die Verzögerung in der Reaktionszeit muss hier mit einer Abweichung von  $\pm 1$  Sekunden gerechnet werden.

#### 3.3.3.4 Werbung

Die Werbung, welche bei den einzelnen Portalen unterschiedlich gehandhabt wird, wurde statistisch nicht erfasst, jedoch wurde dort ein Kommentar dazu abgegeben, wo die Werbung zu einem negativen Kundenerlebnis führt.

### 3.4 Messungen

#### 3.4.1 Qualität

##### 3.4.1.1 Übersicht

Das Kundenerlebnis beim Betrachten von gestreamten Videos wird durch verschiedenste Faktoren bestimmt. Der wohl wichtigste Faktor ist die eigentliche Bildqualität (Auflösung, Bildwiederholungsrate). Der am offensichtlichsten bemerkbare Faktor sind auftretende Darstellungsfehler. Bei Liveübertragungen, wie zum Beispiel von Sportanlässen, kann die Verzögerung gegenüber anderen Verteilkanälen wichtig sein. Welcher Faktor das Kundenerlebnis wie stark beeinflusst, hängt stark vom Anwendungsfall bzw. Programminhalt ab. Dazu gehören auch Werbeeinblendungen, welche das Kundenerlebnis auch bei absolut perfekter Übertragungsqualität sehr ins Negative beeinflussen können.

Die Analyse beschäftigt sich mit messbaren Werten, welche erhoben und verglichen werden können. Dabei wird zwischen quantitativer und qualitativer Messung unterschieden. Die Qualitätswahrnehmung kann sich je nach Benutzer ändern. Diese ist subjektiv und kann am besten in Benutzertests erhoben und überprüft werden. Deshalb wird für die Analyse das Augenmerk vor allem auf quantitative Messwerte gelegt. Aus diesen, möglichst vielen, Werten werden anschliessend Auswertungen durchgeführt.

##### 3.4.1.2 Auflösung

Das Standardauflösungsverhältnis für Video beträgt heutzutage 16:9, womit man nun aus der Angabe der Höhe direkt die Breite berechnen kann. Die drei häufigsten angetroffenen Auflösungen sind in Tabelle 1 wiedergegeben:

Breite	Höhe	Verhältnis	Bezeichnung
1920	1080	16:9	Full HD
1280	720	16:9	HD
640	360	16:9	SD

Tabelle 1: Am häufigsten angetroffene Auflösungen

Während diese Auflösungen von fast allen Videoportalen angeboten werden, gibt es bei YouTube und Netflix einige Videoangebote, welche über eine grössere Auflösung als Full HD verfügen. Für eine vergleichbare Analyse wurde deshalb ein Auflösungsrahmen von 180p bis 1080p festgelegt und berücksichtigt. Nach Steam-Hard- & Softwareumfrage (weltweit) [18] und der Statistik von StatCounter (Schweiz) [19] ist Full HD die momentan am meisten auf Computern und Laptops verwendete Bildschirmauflösung. Somit ist mit diesem Rahmen ein Grossteil der Benutzer abgedeckt und die Aussagefähigkeit der Analyse nicht eingeschränkt.

Da die meisten der Videoportale keine zusätzlichen Auflösungen anbieten, spielt es in Bezug auf die Auflösung keine signifikante Rolle, ob der Inhalt auf dem Computer oder auf mobilen Geräten, speziell Mobiltelefone, konsumiert wird, denn moderne Videostream Übertragungssysteme passen die Auflösung an die aktuellen Kanaleigenschaften an. Die folgende Auflösungsskalierung konnte bei verschiedenen Übertragungsraten beobachtet werden. Die x-Achse beschreibt die verwendeten Mbit/s während die y-Achse die Auflösungshöhe wiedergibt.

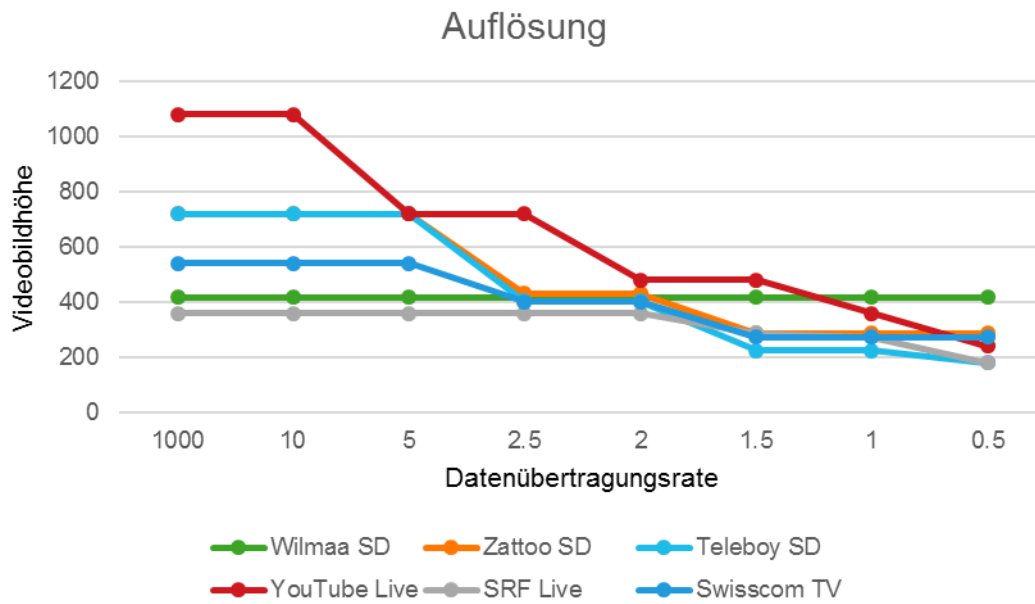


Abbildung 19: Auflösungen von Live Videostreams

Nebst YouTube Live bietet keine der, in dieser Arbeit untersuchten, Live Videoportalen eine Full HD Auflösung an. Am nächsten zusammen liegen die Streams bei 2 Mbit/s, wobei YouTube Live auch hier eine etwas bessere Auflösungsstufe anbietet.

Alle Portale bieten eine Auflösungsstufung an, ausser Wilmaa, welche für jede Datenübertragungsrate nur ihre Standardauflösung von 882x418 anbietet. Dies hat zur Folge, dass die Auflösungsqualität auch bei voller Übertragungsrate schlecht ist und es ab 1.5 Mbit/s zu sehr vielen Unterbrechungen kommt.

Sowohl Zattoo als auch Teleboy liefern eine HD Auflösung von 1280x720 bei hoher Datenübertragungsrate und skalieren problemlos bei ändernder Übertragungsrate.

Interessant ist, dass sowohl Teleboy als auch SRF Live minimalste Auflösungen von 380x180 anbieten. Obwohl YouTube eine minimale Auflösungsstufe von 256x144 anbietet, wurde diese auch durch das Verringern der Übertragungsrate auf 0.5 Mbit/s nicht benutzt, sondern immer noch die Auflösung der höheren Stufe geliefert.

Die Dominanz von YouTube Live in dieser Grafik hat verschiedene Gründe. YouTube verwendet das QUIC Protokoll und hat mit diesem UDP basierten Protokoll einen grossen Vorteil über alle anderen Portale. Mit der Grösse von Google und der weltweiten Benutzer wurden Millionen in die Optimierung von YouTube gesteckt, da können kleinere Schweizer Firmen natürlich nicht mithalten. Diese Optimierungen beginnen bei der Anwendung von Encoder Optionen, dem Code zum Laden und Anzeigen der Webseite und gehen bis hin zu Details von guten Peering mit Internet Service Provider (ISP) und nahen Content Delivery Networks (CDN) [20].

### 3.4.1.3 Bildwiederholungsrate

Die Bildwiederholungsrate (Framerate) zeigt an, wie viele vollständige Frames pro Sekunde aufgebaut und angezeigt werden können. Normalerweise wird diese dementsprechend mit Hertz (Hz) angegeben. Oft wird jedoch auch die Abkürzung FPS (Frames Per Seconds) verwendet.

### 3.4.1.4 Bitrate

Während die Auflösung nur aussagt wie viele Bildpunkte (Pixel) an Videomaterial geliefert werden, so kann alleine damit die konkrete Qualität noch längst nicht bestimmt werden, denn jeder Videostream ist mit einem verlustbehafteten Codec komprimiert. Ohne Komprimierung wäre der gesamte Datenverkehr viel zu gross und da die Unterschiede visuell praktisch nicht bemerkbar sind, macht es keinen Sinn Videostreams nicht zu komprimieren.

Die Bitrate gibt an, wie viel Bits pro Sekunde an Videomaterial verwendet werden, um das Video darzustellen. Unter der Annahme, dass ein Bildpunkt jeweils 24 Bits verwendet, lässt sich die nicht komprimierte, maximale Bitrate folgendermassen berechnen:

$$\text{Bitrate} = \text{Bildwiederholungsrate} * \text{Breite} * \text{Höhe} * 24 \text{ Bits}$$

Formel 1: Berechnung nicht komprimierter, maximaler Bitrate

Dabei wird die Bildwiederholungsrate in Hertz angegeben und die Breite und Höhe entspricht der Auflösung des Videostreams. Durch die Kompression sinkt die Bitrate natürlich sehr stark und kann nicht direkt berechnet werden, sondern muss durch Analyse der Datenübertragungsratenutzung oder des Videostreams in Erfahrung gebracht werden.

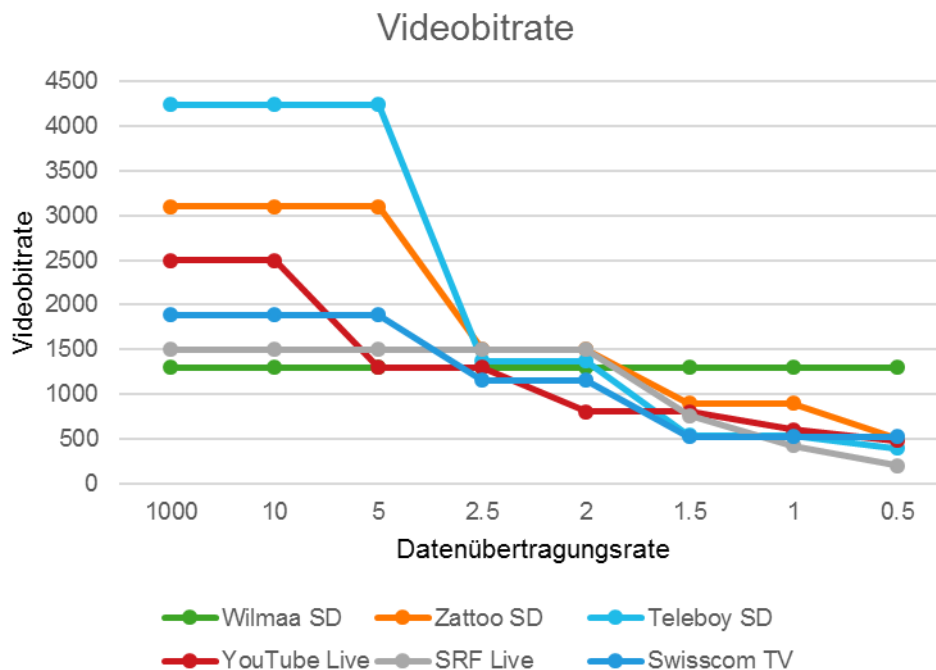


Abbildung 20: Bitraten von Live Videostreams

Gleich wie bei der Analyse zur Auflösung bietet Wilmaa keine Bitrateskalierung an. Sowohl bei hoher Datenübertragungsrate als auch bei tiefer bleibt die Bitrate gleich, weshalb es nun auch nicht überraschend ist, dass der Stream unter 1.5 Mbit/s starke Probleme hat, da die Bitrate mit 1.3 Mbit/s die gesamte Datenübertragungsrate oder sogar mehr benötigen würde, um den Videostream ohne Unterbruch wiederzugeben.

Sehr überraschend ist, dass Teleboy eine überaus hohe Bitrate von bis zu 4240 Kbit/s liefert. Dies ist insbesondere interessant, da Zattoo die gleiche Auflösung für die gegebene Datenübertragungsrate verwendet. Es zeigt sich somit, dass die Qualität bei gleicher Auflösung besser sein kann, da die Bitrate, d.h. die Gesamtbildinformation, unterschiedlich ist.

Die Kurve von YouTube fällt doch etwas aus dem Konzept und dies sogar noch viel stärker wenn die gemessene Bitrate mit der berechneten, nicht komprimierten Bitrate verglichen wird.

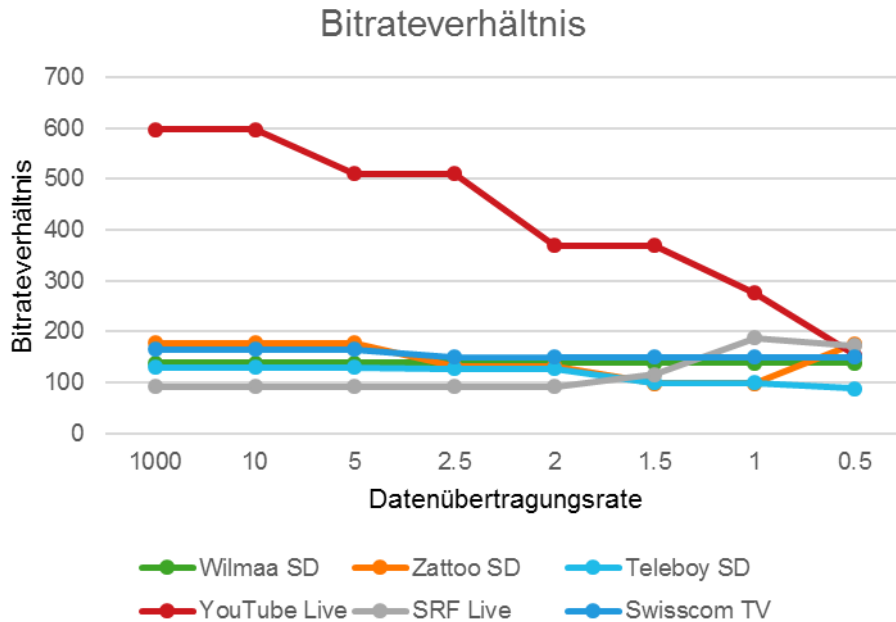


Abbildung 21: Bitraten Verhältnis von Live Videostreams

Der Grund für diesen grossen Unterschied liegt zum einen daran, dass die YouTube Streamer selbst eine Bitrate festlegen und diese nicht zwingend für die gegebene Auflösung optimiert ist. YouTube schlägt zwar eine Skalierungskurve vor [21], aber der Benutzer kann auch ein Video mit Full HD Auflösung aber einer tiefen Bitrate an YouTube schicken. Andere Videoportale haben ihre Streams für die optimale Bitrate zur Auflösung angepasst. Dies erklärt jedoch nicht, wieso das Verhältnis über die gesamte Kurve so hoch ist. Beim genaueren Analysieren das Videostreams, zeigte sich, dass die von YouTube vorgegebene Bildwiederholungsrate von 30 Hz zwar verwendet wird, aber das Originalvideo wohl mit einer tieferen Bildwiederholungsrate von 25 Hz gedreht wurde, denn jedes sechste Frame ist ein Duplikat. Obwohl nun die berechnete Bitrate um ein Fünffaches ansteigt, so kann die effektive Bitrate immer noch auf dem gleichen Niveau bleiben.

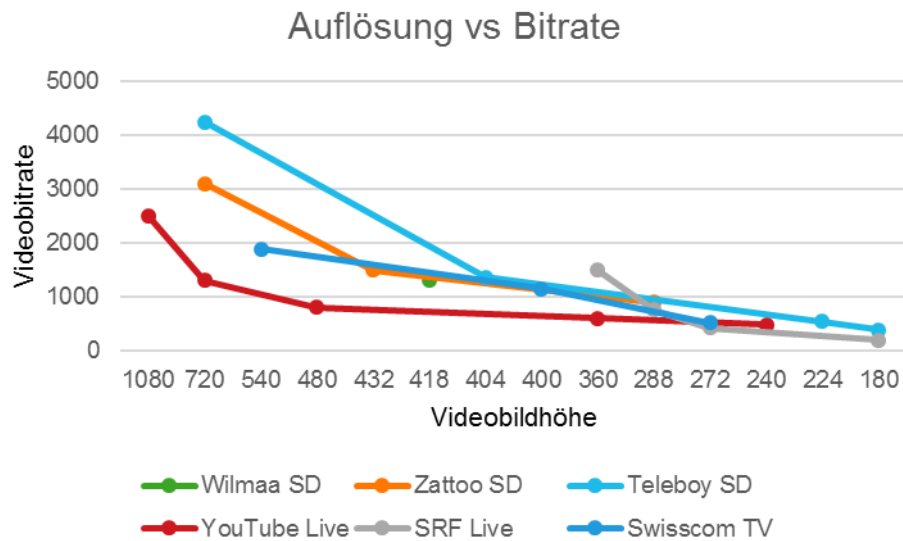


Abbildung 22: Auflösung im Vergleich zur Bitrate für Live Videostreams

Das Gegenüberstellen von Auflösung (x-Achse) und Bitrate (y-Achse) lässt das Phänomen von YouTube erneut erscheinen. Im Vergleich zu den anderen Portalen setzt YouTube auf tiefere Bitraten bei gleicher oder ähnlicher Auflösung, aber die Skalierung verläuft sehr logarithmisch, insbesondere auch weil YouTube feinere Qualitätsstufen anbietet.

Teleboy scheint nach unten eine recht lineare aber steile Skalierung zu haben.

### 3.4.2 Paketverlust

#### 3.4.2.1 Paketverlust

Beim Streaming wird alle Kommunikation über Netzwerkpakete bewerkstelligt. Nun wurde analysiert wie sich die einzelnen Portale bei einem 5% Paketverlust verhalten. 5% Paketverlust bedeutet, dass jedes 20. Paket verloren geht, was in der heutigen Netzwerkwelt sehr hoch ist. Normalerweise muss man mit einer Packet Loss Rate von unter 1‰ (Promille) rechnen, somit geht weniger als jedes 1000. Paket verloren.

Interessant ist die Erkenntnis, dass auch das Streaming Protokoll eine Rolle beim Verhalten spielt. Denn während DASH sogar bei 15% Paketverlust das Video teilweise noch abspielen kann, es sind nur kleinere Qualitätseinbussen, wie zum Beispiel eine kleinere Auflösung sichtbar, kann HLS mit 10% Paketverlust nicht mehr starten, geschweige denn das Video streamen.

Alle folgenden Erkenntnisse sind mit 5% Paketverlust erfolgt. Dabei muss beachtet werden, dass der Streamaufbau über TCP vollzogen wird. Bei einem hohen Paketverlust kann es vorkommen, dass der TCP Aufbau, d.h. der Three-Way-Handshake, nicht korrekt vollzogen werden kann und deshalb der Stream, unabhängig vom Anbieter, zum Teil auch gar nicht erst geladen wird.

### 3.4.2.2 Auflösung

Die Auflösung verhält sich sehr ähnlich wie im normalen Betrieb. Viele der Eigenschaften und Erkenntnisse aus Abschnitt 3.4.1 sind hier gleich. Im Schnitt ist die Auflösung beim Paketverlust einfach eine bis zwei Stufen kleiner.

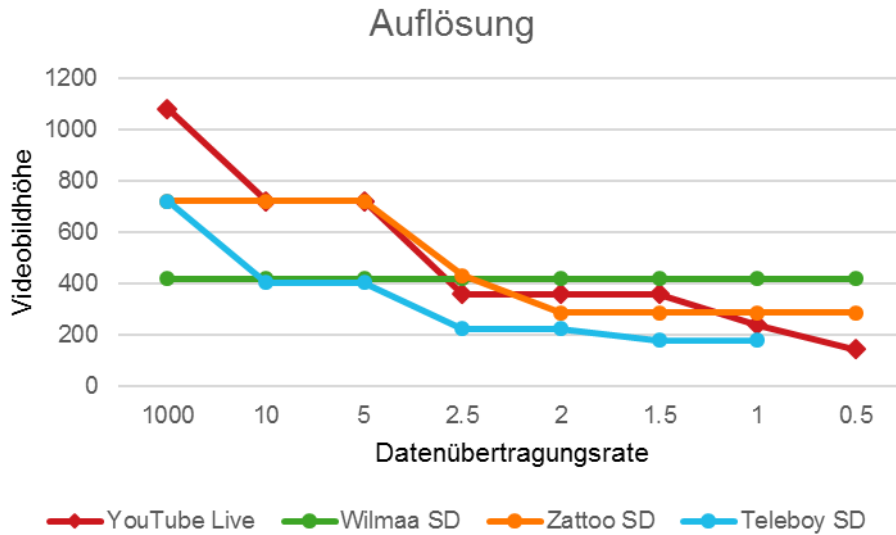


Abbildung 23: Auflösung von Live Videostreams

Die folgende Auflösungskalierung konnte bei verschiedenen Datenübertragungsraten beobachtet werden. Die x-Achse beschreibt die verwendeten Mbit/s während die y-Achse die Auflösungshöhe wiedergibt.

Obwohl YouTube als auch Netflix Ultra HD Auflösungen anbieten, das heisst bis zu einer Auflösung von 3840x2160 (4K) oder sogar 7680x4320 (8K), wurden diese nie gewählt, weil die Labor Computer keine solch hohe Auflösung besaßen.

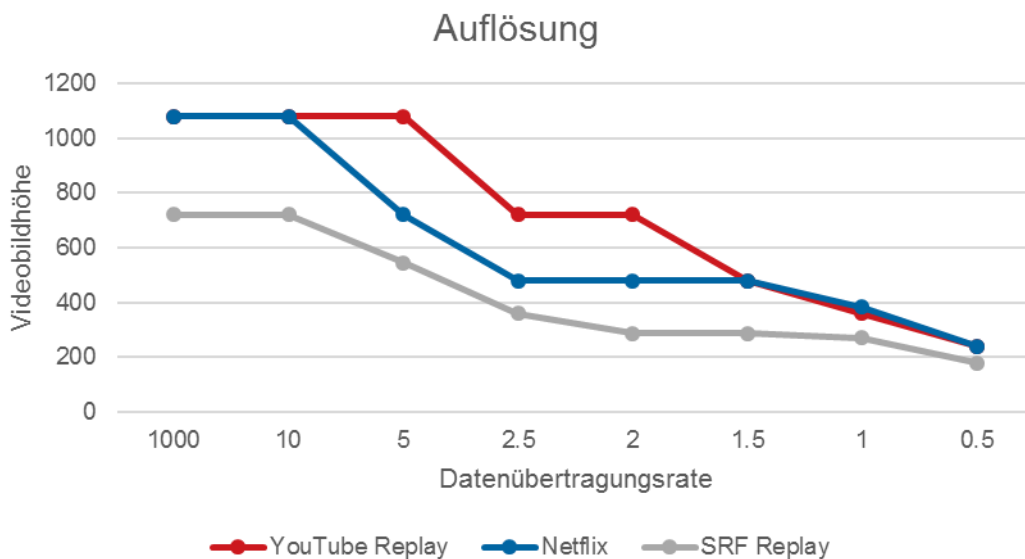


Abbildung 24: Auflösungen von Replay Videostreams

Die Skalierung verhält sich bei allen drei Videoportalen relativ ähnlich, wobei SRF bei kleiner werdender Datenübertragungsrate früher die Auflösung zurücknimmt.



### 3.4.2.3 Bitrate

Die Bitrate verhält sich sehr ähnlich wie im normalen Betrieb. Viele der Eigenschaften und Erkenntnisse entsprechen den Daten aus Abschnitt 3.4.1. Im Schnitt ist die Bitrate beim Paketverlust einfach eine bis zwei Stufen kleiner.

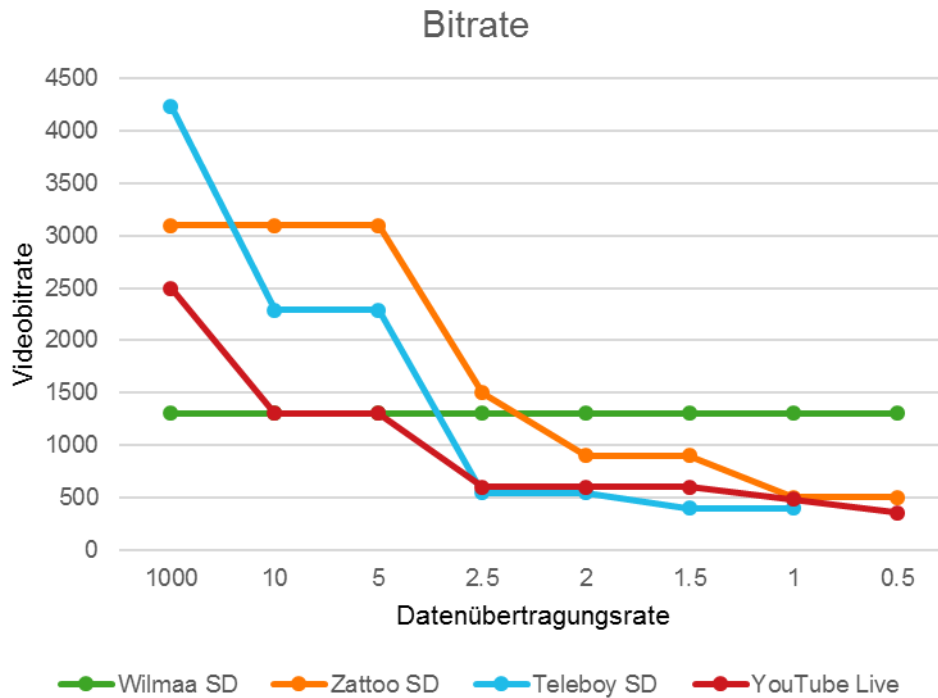


Abbildung 25: Bitraten von Live Videostreams

Erkennbar ist hier, dass Teleboy zwar mit der höchsten Bitrate startet, aber Zattoo schlussendlich die beste Bitrate pro Stufe aufweist.

Im folgenden Diagramm wird mit den Videoportalen Zattoo und Teleboy anschaulich, wie sich die Qualitätseinbussen in Bezug auf die Auflösung, bei einem 5% Paketverlust verhält.

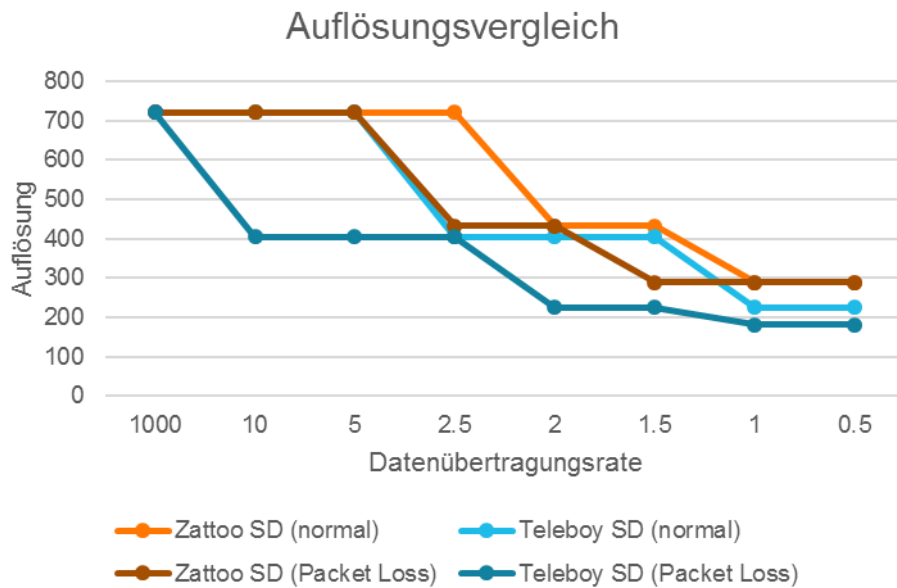


Abbildung 26: Auflösungsvergleiche Normal vs. Paketverlust

Auch hier brilliert YouTube wieder mit einer sehr guten Kompression und daraus resultierendem Bitrate Verhältnis.

Bei Replay Videostreams ist die Bitrate fast noch wichtiger als bei Live Videostreams, denn diese können nun bis ins letzte Detail optimiert werden, worüber insbesondere auch Netflix sehr gerne berichtet. [22]

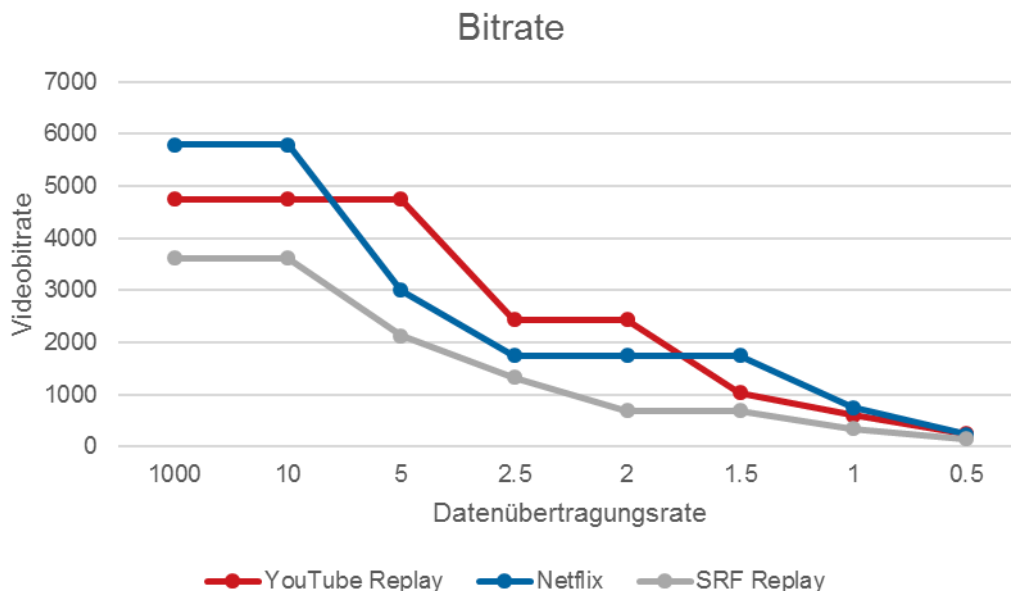


Abbildung 27: Bitraten von Replay Videostreams

Im Vergleich zu Abbildung 24, lässt sich hier schön der Grund erkennen, weshalb YouTube Replay auch bei 5 Mbit/s noch eine Full HD Auflösung liefern kann. Netflix verwendet dafür nämlich eine Bitrate von über 5 Mbit/s, während YouTube Replay knapp unter die 5 Mbit/s Grenze fällt.

Dass YouTube bei einer Datenübertragungsrate von 2 Mbit/s trotzdem noch Videos mit einer Bitrate von 2.5 Mbit/s abspielen kann, ist wohl nur dank Buffering und einer durchschnittlich tieferen Bitrate zu erklären.

Nicht überraschend ist das Annähern der Kurven bei tieferer Datenübertragungsrate, denn dort gibt es nur noch sehr wenig Spielraum was die Videoqualität angeht.

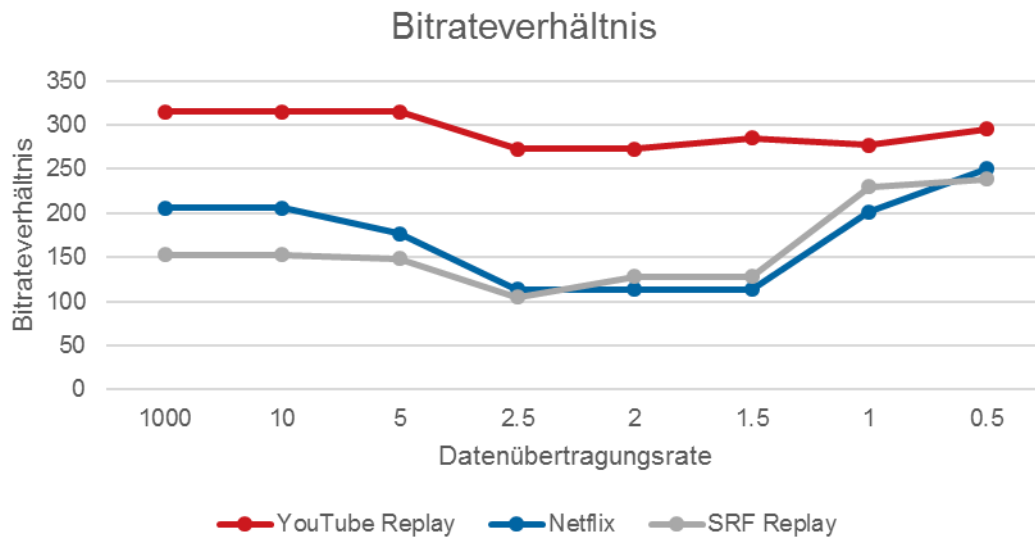


Abbildung 28: Bitraten Verhältnis für Replay Videostreams

Das Interessante an diesem Diagramm ist, dass die Komprimierung bei höheren Qualitäten wohl etwas stärker greift als in der Region für 1.5 bis 2.5 Mbit/s und dann am Ende doch wieder stark zunimmt.

Wie bereits bei den Live Videostreams festgestellt, so bietet YouTube im Allgemeinen eine kleinere Bitrate an, was zu einem höheren Verhältnis führt.

### 3.4.3 Werbung

Die meisten hier analysierten Live-Videoportale stellen Benutzern kostenlos Standardqualität zur Verfügung. Nur für HD müssen Abos gelöst und bezahlt werden. Um sich trotz freiem Inhalt zu finanzieren, zeigen alle, bis auf SRF, Werbung. Dabei verhalten sich die einzelnen Portale jedoch sehr unterschiedlich. Von YouTube, mit eher zufälligen Werbeeinblendungen, über Swisscom TV Air, mit einer Werbung pro Einschaltung, zu Teleboy, mit zwei Werbungen von 15-30 Sekunden Länge, welche bei jedem Senderwechsel oder aktualisieren der Website blockieren und sogar erscheinen falls ein Unterbruch zu lange dauert. Zu allem Übel werden diese Werbungen nicht skaliert, was bedeutet, dass umso länger gewartet werden muss, desto schlechter die Verbindung ist. In Bezug auf die User Experience ist dieses Verhalten nicht befriedigend. Dadurch wird klar, dass YouTube die beste Handhabung mit Werbung hat und Teleboy die schlechteste.

### 4 Entwicklung eines Video Stream Analysers

#### 4.1 Technischer Aufbau

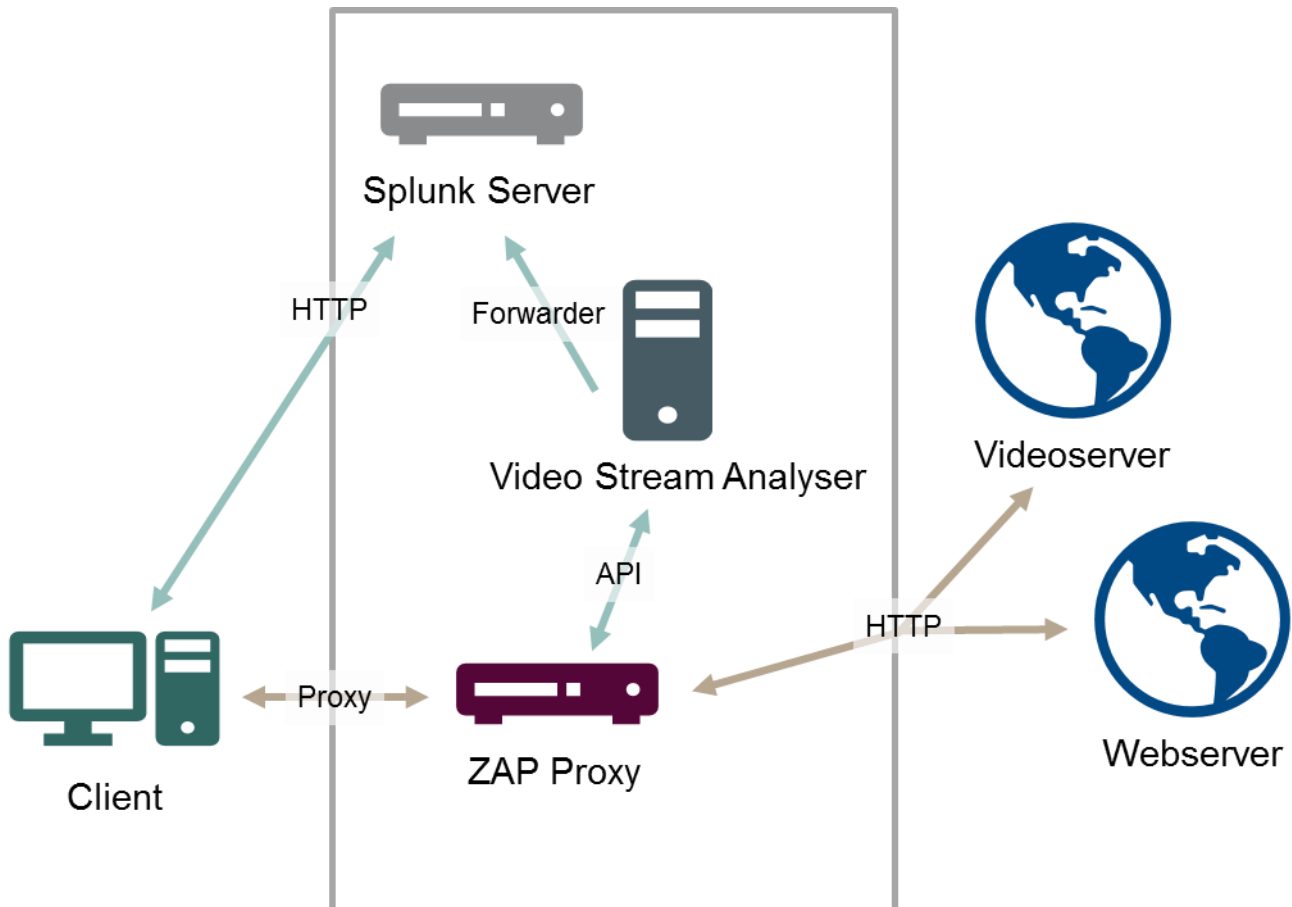


Abbildung 29: Infrastruktur des Video Stream Analysers

Der Videoplayer wird von der Webseite des Videoportals geladen. Danach fordert der Player das Manifest an und liest aus diesem, welche Videostreams vom Server zur Verfügung gestellt werden.

Der ZAP Proxy sitzt zwischen den Servern und dem Client und nimmt den gesamten Traffic auf.

Der Video Stream Analyser holt sich über die ZAP API die nötigen Informationen vom ZAP Proxy. Falls ein Videostream gefunden wurde, leitet der Analyser die daraus extrahierten Metadaten an Splunk weiter.

Splunk nimmt schlussendlich alle Daten auf. Der Benutzer des Systems oder ein zusätzlicher Analyst kann die Daten in Splunk dann mit verschiedensten Queries herausfiltern und visualisieren.

### 4.2 Struktur

Der Video Stream Analyser besteht aus drei Teilen. Das Command Line Interface behandelt die Eingabe über ein Terminal, verarbeitet die Parameter und startet entweder die Applikation oder gibt den Programm Hilfetext zurück. Das Hauptprogramm läuft dann innerhalb der Streamzap Klasse ab, welche zuständig für das Initialisieren der ZAP API und das Vorbereiten aller Dienste ist.

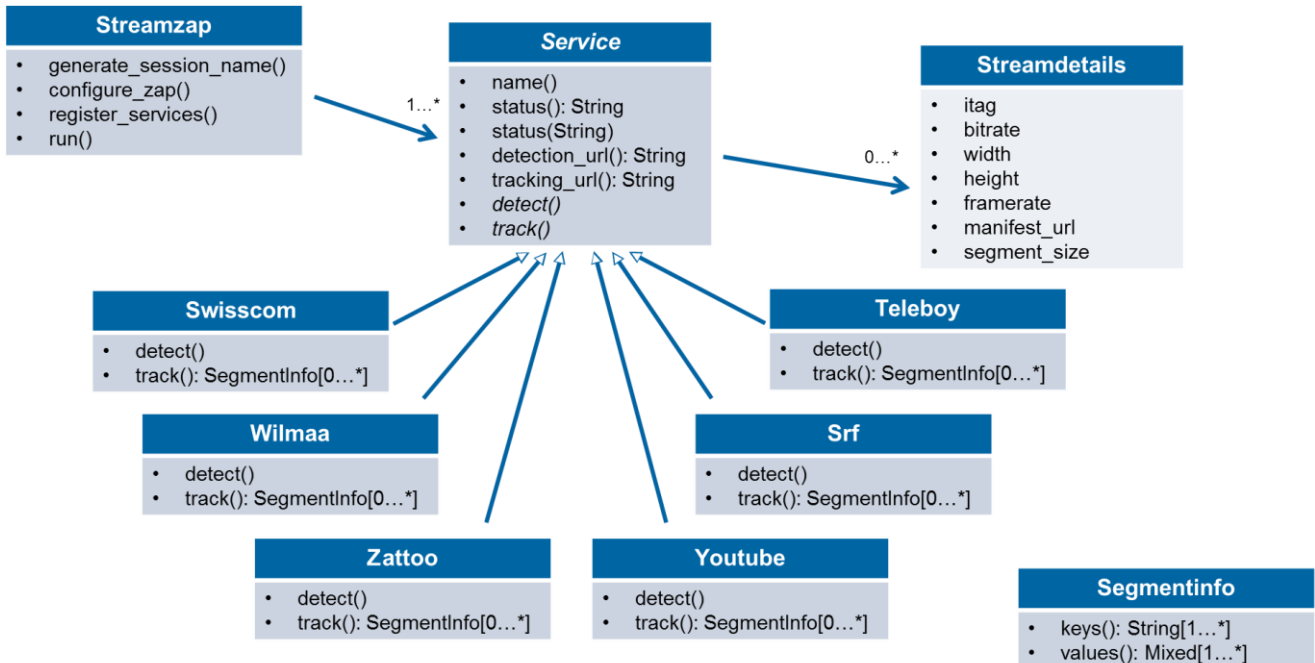


Abbildung 30: Struktur des Video Stream Analysers

Und als dritter Teil gibt es die verschiedenen Serviceklassen (Swisscom, Teleboy, Wilmaa, Srf, Zattoo, Youtube), welche von einer Basisklasse erben, um gemeinsame Grundfunktionalitäten und ein einheitliches Interface zu haben. Falls ein weiteres Videoportal unterstützt werden soll, muss dank des modularen Aufbaus nur eine neue Serviceklasse erstellt und die beiden Funktionen *detect()* und *track()* implementiert werden.

Die Hilfsklasse Streamdetails wird verwendet um Informationen zu den gefundenen Streams innerhalb eines Dienstes zu speichern. Die Segmentinfo Klasse kommt beim Speichern der gefundenen Segmente zum Einsatz und wird als Kommunikationsmittel zwischen den Serviceklassen und der Streamzap Klasse verwendet.

### 4.3 Funktionen

Die zentrale Funktionsweise des Video Stream Analysers läuft folgendermassen ab: So lange es zu keinem Interrupt via Tastatur des Benutzers kommt, wird über alle Dienste iteriert und für jeden den Status überprüft. Ein Dienst kann entweder in der Detection oder der Tracking Phase sein.

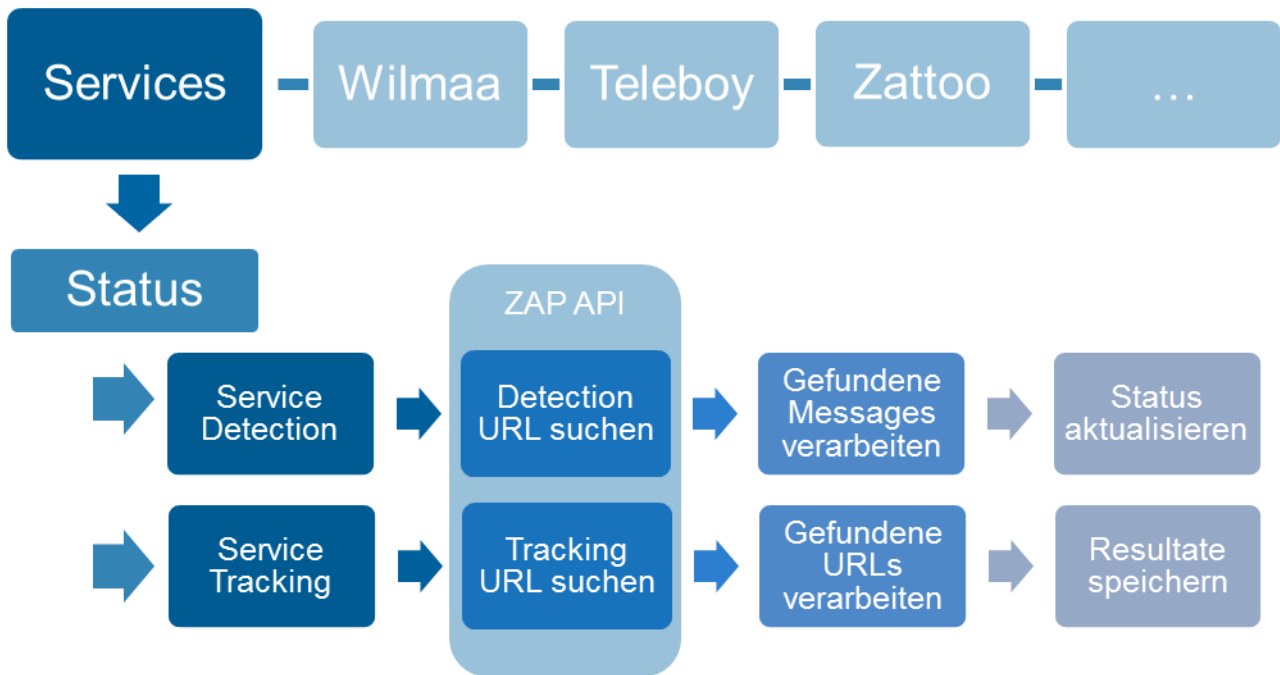


Abbildung 31: Interner Ablauf des Video Stream Analysers

#### 4.3.1 Service Detection

Jede Dienstklasse definiert einen Regular Expression (Regex) String einer URL, welche beim erstmaligen Starten eines Videostreams erwartet wird. Bei einigen Videoportalen entspricht diese initiale URL dem Abruf des Manifests. Bei anderen wie zum Beispiel YouTube, wird der Dienststart beim Öffnen einer bestimmten Seite festgestellt. Falls die Detection URL tatsächlich im Netzwerkverkehr gefunden wird, verarbeitet der Dienst den Inhalt der HTTP Antwort (Response Body). Wenn die Verarbeitung ergibt, dass es sich in der Tat um einen aktiven Videostream des ausgewählten Videoportals handelt, so wird der Status von Detection auf Tracking geändert.

#### 4.3.2 Service Tracking

Die Funktionsweise des Service Tracking ist sehr ähnlich. Während der Detection Phase wird eine Tracking URL gesetzt, welche entweder statisch einprogrammiert ist oder dynamisch mit weiteren Informationen aus dem Manifest erstellt wird. Bei jeder Applikationsiteration wird die Tracking URL im Netzwerkverkehr gesucht. Falls es zu einer Übereinstimmung kommt, werden die Videostream Merkmale extrahiert und die Resultate für die weitere Verarbeitung in einer CSV-Datei gespeichert.

### 4.3.3 Merkmale

Da es bei der Auswertung für Splunk keine Rolle spielt wie viele Daten gesendet werden, wird versucht möglichst viele Informationen aus den Videostreams herauszufischen.

Bezeichnung	Erklärung
timestamp	Zeitpunkt wann die Daten erfasst wurden
session	Name der aktuellen Session
service	Name des verwendeten Videoportals
protocol	Verwendetes Streaming Protokoll
bitrate	Bitrate des Videostreams
width	Videoauflösungsbreite
height	Videoauflösungshöhe
framerate	Bildwiederholungsrate des Videostreams
segmenturl	Die URL des heruntergeladenen Segments
segmentsize	Die Grösse des Segments
itag	YouTube verwendet dieses Tag zur Identifikation vom Videotyp

**Tabelle 2: Videostream Merkmale**

Die Merkmale in Tabelle 2 werden erfasst, sind aber nicht bei allen Videoportalen vorhanden. Wenn von einem Videoportal ein Merkmal nicht erkannt wird, so wird der Platzhalter «N/A» verwendet.

## 5 Auswertung mit Splunk

### 5.1 Splunk Möglichkeiten

Mit Splunk können fast alle Arten von maschinellen Daten gesammelt, indiziert, verarbeitet und ausgewertet werden. In dieser Arbeit wird mit der gratis Version Splunk Light [23] gearbeitet.

### 5.2 Erfassung

Gewünschte Daten können durch den Button «Daten hinzufügen», welcher sich auf der Übersichtsseite befindet, hinzugefügt werden. Für den Datenupload gibt es drei verschiedene Szenarien.

#### Daten hinzufügen

Wie möchten Sie Daten hinzufügen?



Abbildung 32: Splunk Dateneingabe

#### 5.2.1 Manuelles Hochladen

Das einfachste ist eine lokale, strukturierte Datei (wie z.B. ein CSV) manuell hochzuladen. Dabei wird die Datei ausgewählt und anschliessend können ein paar Einstellungen konfiguriert werden. Diese sind der Sourcetype (Standard: Automatisch erkennen), der Host (Standard: Name des Computers) und der Index (Standard: Standard). Die jeweiligen Standards sind empfehlenswert, mit Ausnahme des Index. Hier, wie auch in den anderen zwei Szenarien, ist es besser einen bestimmten Index zu verwenden oder wenn nötig einen neuen zu erstellen. Die Daten in einem Index gehören logisch zusammen und einmal im gleichen Index hinzugefügt, ist es sehr schwer, spezifische Daten wieder daraus zu entfernen oder separat zu behandeln.



### 5.2.2 Quelle überwachen

Das zweite Szenario ist bestimmte Quellen zu überwachen. Hier kann aus möglichen Quellen ausgewählt werden, unter anderen sind dies zum Beispiel Ereignisprotokolle, Netzwerk Port oder Windows-Überwachungen.

Daten hinzufügen

Quelle auswählen Sourcetyp festlegen Eingabeeinstellungen Prüfen Fertig

**Lokale Ereignisprotokolle**  
Erfassen Sie Ereignisprotokolle von diesem Rechner.

**Remote-Ereignisprotokolle**  
Erfassen Sie Ereignisprotokolle von Remote-Hosts. Hinweis: Hierfür wird WMI verwendet und Sie benötigen ein Domänenkonto.

**Dateien und Verzeichnisse**  
Sie können eine Datei hochladen, eine lokale Datei indizieren oder ein ganzes Verzeichnis überwachen.

**HTTP-Ereignissammlung**  
Konfigurieren Sie Token, die Clients verwenden können, um Daten über HTTP oder HTTPS zu senden.

**TCP / UDP**  
Konfigurieren Sie Splunk, um die über einen Netzwerkport gesendeten Daten zu überwachen.

**Lokale Leistungsüberwachung**

Konfigurieren Sie diese Instanz, um Dateien und Verzeichnisse auf Daten zu überwachen. Für das Monitoring aller Objekte in einem Verzeichnis wählen Sie das Verzeichnis aus. Splunk weist allen Objekten innerhalb des Verzeichnisses einen Sourcetyp zu und überwacht diesen. Dies kann zu Problemen führen, wenn das Verzeichnis unterschiedliche Objekttypen oder Datenquellen enthält. Um Objekten im gleichen Verzeichnis mehrere Sourcetypen zuzuweisen, konfigurieren Sie individuelle Dateneingaben für diese Objekte. [Erfahren Sie mehr](#)

Datei oder Verzeichnis?

Unter Windows: C:\apache\apache.error.log oder \\Hostname\apache\apache.error.log.  
Unter Unix: /var/log oder /mnt/www01/var/log.

Whitelist?

Blacklist?

Abbildung 33: Splunk Überwachen

Für die Arbeit wurde die Dateien- und Verzeichnis-Überwachung aufgesetzt und konfiguriert. Dabei wird das Verzeichnis mit den Service-CSVs ausgewählt und dieselben Einstellungen wie im ersten Szenario übernommen. Während man eine Datei gemäss dem ersten Szenario beliebig oft hochladen kann, wird hier bei jeder Quelleneingabe überprüft, ob diese schon existiert. Wurde eine bereits vorhandene Quelle gefunden, wird eine Fehlermeldung ausgegeben.

### 5.2.3 Weiterleiten

Bei dem dritten Szenario kann ein Splunk-Forwarder eingerichtet werden. Zuerst muss der Universal Forwarder von Splunk installiert und konfiguriert werden, anschliessend können gewünschte Forwarder installiert werden.

### 5.3 Auswertung

Die Auswertung erfolgt über die Splunk-eigene Suchsprache. Hier können alle gewünschten Daten angezeigt werden (je grösser die Datenmenge, desto länger dauert der Anzeigebau).

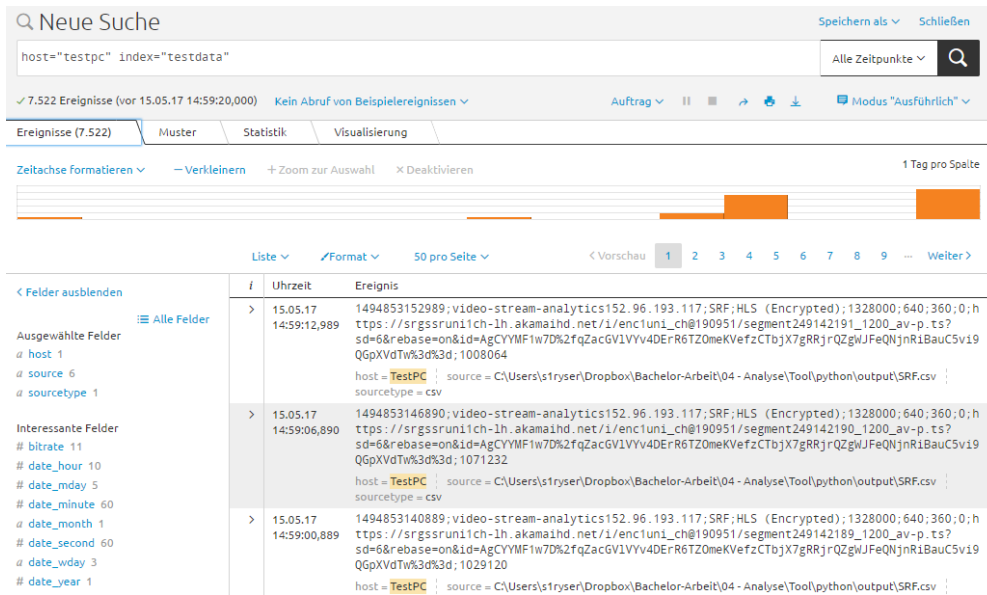


Abbildung 34: Splunk Suche

Jedes Ereignis ist ein separater Event, welcher genauer untersucht werden kann.

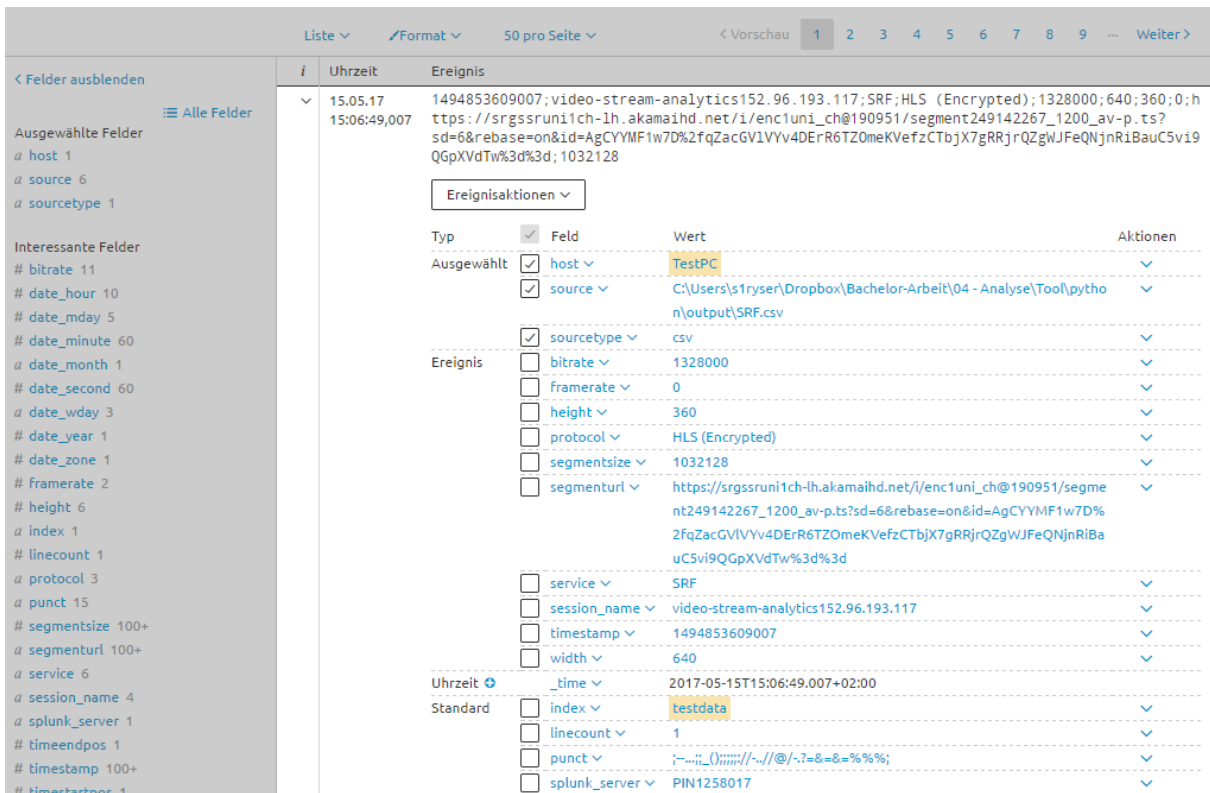


Abbildung 35: Splunk Ereigniseinsicht

In Abbildung 35 sieht man die Parameter, welche gemäss Anwendung dieser Arbeit mit dem Python Skript erfasst und von Splunk in Echtzeit automatisch eingefügt und indiziert wurden. Gelb hervorgehoben werden Eigenschaften, welche in der Suche vorkommen.

```
host="testpc" index="testdata" | eventstats count(service) as Total | stats count(service) as Count, values(Total) as Total  
by service | eval percent=round((Count/Total)*100,1) | eval service= service.", ".percent.%" | table service, Count
```

Abbildung 36: Splunk Suchbefehle

Es gibt fast unzählige Möglichkeiten Daten zu suchen und durch Verschachtelung verschiedener Befehle, auszuwerten. Die einzelnen Befehle werden durch „|“ voneinander abgetrennt. Da es viel zu viel wäre, alle Suchbefehle und die jeweiligen Optionen durchzugehen, werden einfach die grundlegenden Suchbefehle gezeigt. Am wichtigsten ist die grobe Einschränkung der Daten durch Host- und Indexangaben. Hier wird die Bedeutung eines sinnvollen Index wieder deutlich. Neben dieser groben Dateneinschränkung sind folgende Befehle in dieser Arbeit wichtig:

**Stats:** Berechnet die Gesamtstatistik über die gesuchten Ergebnisse wie zum Beispiel den Durchschnitt, die Anzahl und die Summe.

**Eventstats:** Fügt Summenstatistiken zu allen Suchergebnissen hinzu. Ist ähnlich wie der Befehl stats. Der Unterschied ist, dass mit dem Eventstats-Befehl Ergebnisse inline zu jedem Ereignis hinzugefügt werden und zwar nur dann, wenn die Summe für dieses Ereignis relevant ist.

**Eval:** Evaluert einen Ausdruck und setzt den resultierenden Wert in ein definiertes Feld. Wenn dieses Feld mit einem bereits vorhandenen Feldnamen übereinstimmt, überschreibt er den vorhandenen Feldwert mit den Ergebnissen des eval-Befehls. Der eval-Befehl wertet mathematische, string- und boolesche Ausdrücke aus.

**Table:** Gibt eine Tabelle mit den gewünschten Feldern. Die Spalten werden dabei gemäss der Argumenten-Reihenfolge angezeigt und jede Reihe zeigt einen Event mit den jeweils zugehörigen Werten.

### 5.4 Visualisierung

Für eine gewünschte Visualisierung ist der Chart-Befehl wichtig. Durch diesen wird Splunk mitgeteilt, dass die Auswertung in einem Diagramm visualisiert werden soll. Mit dem GUI von Splunk können einzelne Diagramme schnell erstellt werden. Sind aber bestimmte Einstellungen gewünscht oder sollen die einzelnen Diagramme in einem Dashboard sinnvoll dargestellt werden, müssen diese im XML Format beschrieben werden. Hauptteil jedes Diagramms bleibt die vorher bestimmte Suche.

```

</chart>
</panel>
<panel>
<title>Summe von Segmentgrösse nach Service</title>
<chart>
<search>
<query>index="testdata" | stats sum(segmentsize) by service</query>
<earliest>0</earliest>
<latest></latest>
</search>
<option name="charting.chart">pie</option>
<option name="charting.chart.showPercent">>true</option>
<option name="charting.fieldColors">{"SRF":0xA8A8A8, "Swisscom TV Air":0x2299DD, "Teleboy":0x1F8BE8, "Wilmaa":0x3DA526, "YouTube":0XCC181E, "Zattoo":0xFF7700}
</option>
</chart>
</panel>
</row>
<row>
<panel>
<title>Durschnittliche Segmentgrösse</title>
<chart>
<search>
<query>index="testdata" | where height!="N/A" and height!=0 | chart avg(segmentsize) by height, service</query>
<earliest>0</earliest>
<latest></latest>
</search>
<option name="charting.axisTitleX.text">Auflösungshöhe</option>
<option name="charting.axisTitleY.text">Durschnittliche Segmentgrösse</option>
<option name="charting.chart">bar</option>
<option name="charting.fieldColors">{"SRF":0xA8A8A8, "Swisscom TV Air":0x2299DD, "Teleboy":0x1F8BE8, "Wilmaa":0x3DA526, "YouTube":0XCC181E, "Zattoo":0xFF7700}
</option>
</chart>
</panel>
</row>
</row>

```

Abbildung 37: Splunk Dashboard XML

## 5.5 Splunk Kurs

Der Splunk Kurs ist für Teilnehmer gedacht, welche sich für Datenanalyse und insbesondere Splunk, als Software zur Datenauswertung, interessieren. Er ist als Weiterbildungskurs im Bereich Big Data gedacht und damit für eine überschaubare Klasse mit motivierten Studenten konzipiert.

### 5.5.1 Kursinhalt

- Lerninhalt:** Ausgewählte Themen von Datenanalysierung und -auswertung kennen und die Software Splunk als entsprechendes Werkzeug anwenden können.
- Vorausgesetzte Kenntnisse:** Keine
- Skriptablage:** Alle Skripte und Dokumentationen sind auf Moodle verfügbar.
- Lernziele:** Der Kursteilnehmer
- kennt verschiedene Möglichkeiten zur Datenanalyse
  - kennt verschiedene Möglichkeiten zur Datenauswertung
  - kennt die Grundlagen für Videostreaming
  - versteht das Setup des Kurses
  - kennt Splunk und kann damit gezielt Daten indexieren, analysieren, auswerten und visualisieren
  - kennt die besten Diagramme für gewünschte Visualisierungen
  - kann grundlegendes XML (für Splunk Dashboards) programmieren
- Kursart:** Vorlesung inkl. Übungen verteilt über zwei Wochen.

### 5.5.2 Use Cases

#### 5.5.2.1 Aktoren

Aktoren	Beschreibung
Student	Der Student ist in dem Kurs angemeldet und nimmt am Kursinhalt teil, gemäss dem Standard Use Case 01. Zusätzlich besteht für den Studenten die Möglichkeit zuhause Teile des Kurses zu vertiefen, gemäss dem Use Case 02.
Kursleiter	Der Professor leitet den Kurs. Dadurch ist er verantwortlich für den Inhalt desselben. Die Rolle verwaltet alle benötigten Utensilien, wie zum Beispiel Server und Computer und hat Zugriff auf die Datenauswertung auf dem dedizierten Server.

Aktoren	Beschreibung
Kursteilnehmer	In der Rolle als Kursteilnehmer sind alle Studenten und Kursleiter des Kurses definiert.
Administrator	Der Administrator ist verantwortlich für den Aufbau der benötigten Infrastruktur und die Unterhaltung derselben. Kann auch die Rolle des Kursleiters innehaben.

Tabelle 3: Aktoren

## 5.5.2.2 Use Case Beschreibung

Für den Splunk Kurs wurden die folgenden zwei Use Cases definiert.

### Use Case 01: Kursteilnahme (HSR)

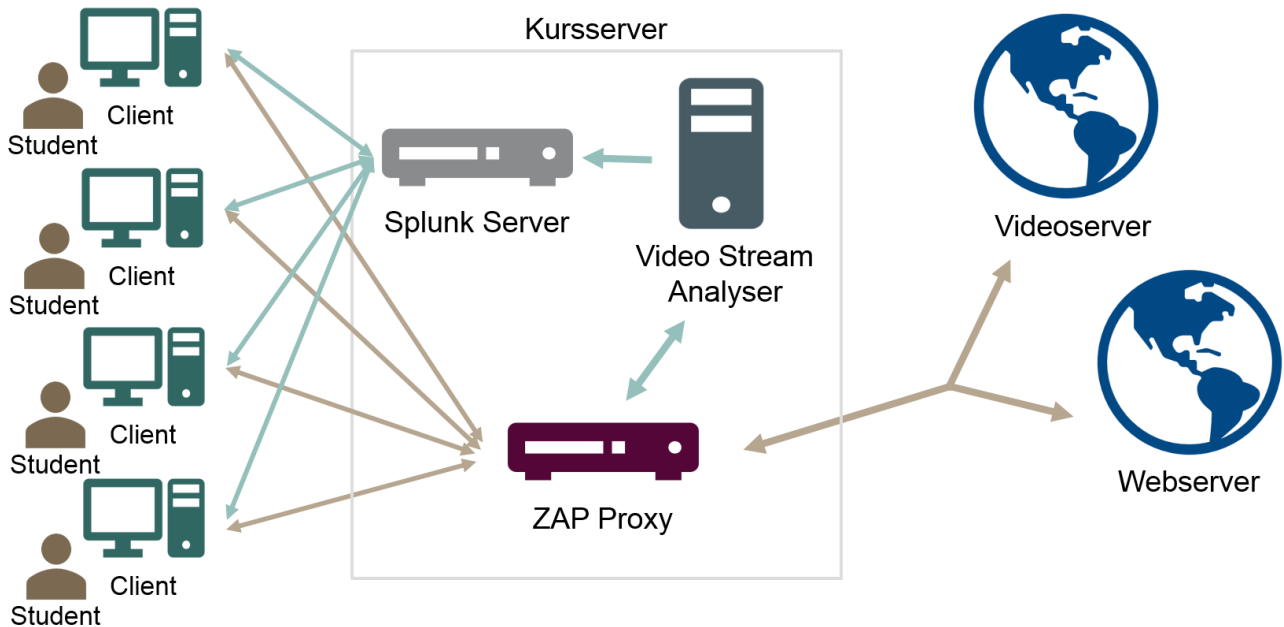


Abbildung 38: Splunk Kurs Setup UC01

Das Herzstück des Kurssetups, wie in Abbildung 38 gezeigt, ist ein (oder mehrere Server), auf welchem der ZAP Proxy, der Video Stream Analyser und der Splunk Server läuft. Durch die Führung aller Streamverbindungen über den zentralen Proxyserver wird eine grössere Datenmenge analysiert und gespeichert, somit werden interessantere und aussagekräftigere Auswertungen ermöglicht.

Der ZAP Proxy und das Streamzap Skript sind für die Studenten in Bezug auf den Splunk Kurs nicht von Interesse. Deshalb wird den Teilnehmern im Kurs ein ZAP Proxy zur Verfügung gestellt auf den sie sich verbinden können. Nach Installation des ZAP Zertifikates und dem Einrichten der Netzwerkumgebung auf den entsprechenden Proxy kann gleich mit dem Videostreaming begonnen werden. Alle Parameterdaten werden für jeden einzelnen Stream vom Video Stream Analyser erfasst und in bestimmten Dateien festgehalten und für den Splunk Server bereitgemacht. Diese Dateien werden auf eine, für alle Teilnehmer lesezugänglichen Dateiablage gespeichert. Anschliessend können die erfassten Daten regelmässig von Splunk indexiert werden und sind somit bereit für gewünschte Auswertungen. Für den Kursleiter werden alle Daten auf dem Splunk Server indexiert, ausgewertet und visualisiert.

Die Studenten haben für ihre Ersterfahrung mit Splunk zwei Optionen zur Verfügung. Bei der einfacheren und empfohlenen Option wird den Studenten ein Splunk-Login auf dem Kursserver angeboten, auf dem sie den Kursinhalt mit allen möglichen erfassten Daten anwenden können. Die zweite Möglichkeit besteht darin, dass die Studenten im Kurs mit ihrer individuell installierten Splunk Light Software arbeiten und die Daten auf ihrem persönlichen Gerät indexieren. Mit beiden Optionen lernen sie Splunk kennen und können mit einer beachtlich wachsenden Datenmenge bereits erste Auswertungserfolge erzielen und aussagekräftige Visualisierungen kreieren.

### Use Case 02: Kurswiederholung (Zuhause)

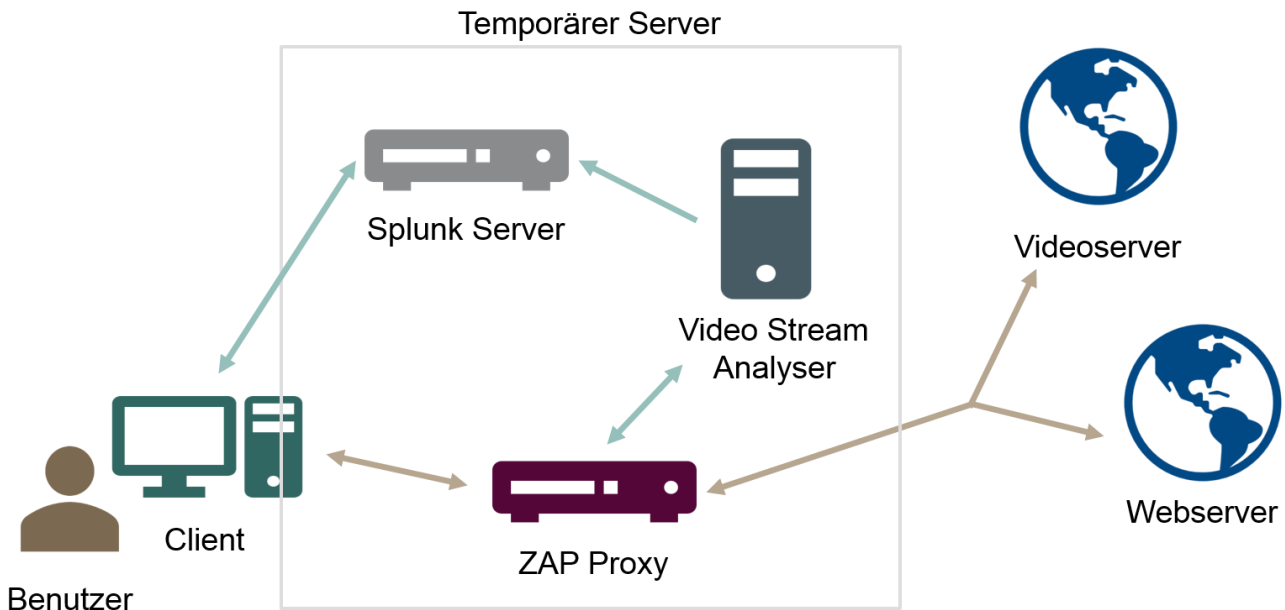


Abbildung 39: Splunk Kurs Setup UC02

Zusätzlich zum Kurssetup an der HSR wird den Studenten die Möglichkeit angeboten, zu Hause ihr Wissen und ihre Fähigkeiten zu vertiefen. Dabei stehen den Studenten zwei Optionen zur Verfügung. Technikbegabte Studenten können das Setup ohne grössere Probleme auf ihrem persönlichen Computer oder Laptop erstellen. Dies bedeutet, dass das gesamte Setup mit der Splunk Light Software, dem ZAP Proxy und dem Streamzap Skript installiert und richtig konfiguriert werden muss. Für Studenten welche die Materie kennen stellt dieses Setup zwar keine grossen Probleme, dafür aber einen Mehraufwand dar. Um den Aufwand niedrig zu halten und auch Studenten, welche keine zusätzlichen Applikationen installieren möchten oder gar nicht die Möglichkeit dazu haben, die gleiche Gelegenheit zu geben, wurde ein Image einer Debian-Umgebung erstellt, welche das Setup bereits installiert und entsprechend konfiguriert hat. Dieses kann auf einem genügend leistungsfähigen Gerät an die Studenten abgegeben werden. Zu Hause müssen die Studenten das Gerät nur noch an den Router anschliessen und alles funktioniert für sie wie im Kurs in den HSR Räumlichkeiten.

Das Debian-Image wurde erstellt, um eine einfache und schnelle Installation auf mögliche neue Geräte zu gewährleisten. Während der Arbeit wurde der fitPC 2i verwendet und getestet, jedoch erwies sich dieser als zu wenig leistungsfähig, um erfolgreich Parameterdaten zu sammeln.

Die Software auf dem fitPC 2i funktioniert zwar beinahe einwandfrei, der Intel Atom Prozessor ist aber zu langsam, vor allem um den Java-basierten ZAP Proxy befriedigend zu betreiben. Deshalb konnten die verwendeten Testgeräte wegen dieser Geschwindigkeitseinbussen keinen Videostream erfolgreich starten. Aus diesem Grunde muss, für eine aussichtsreiche Kurswiederholung zu Hause, ein leistungsstärkeres Gerät benutzt werden.

## Debian Image

Das Image wurde speziell für einen portablen Computer mit einem Intel Chipsatz erstellt, der den Kursteilnehmern mit nach Hause gegeben werden kann. Als Betriebssystem wurde Debian 8.8 (Codename Jessie) verwendet, in Zusammenhang mit der ressourcenschonenden LXDE Oberfläche. Alle für das Setup benötigten Programme sind vorinstalliert und der ZAP Proxy sowie der Video Stream Analyser starten selbstständig nach dem automatischen Login.

### 5.5.3 Kursutensilien

---

Für den Splunk Kurs werden folgende Geräte und Applikationen benötigt.

Die Teilnehmer benötigen:

- Persönlichen Computer, Laptop, Tablet oder Mobiletelefon, welche mit dem Internet verbunden sind und Video streamen können.

Die Kursleiter benötigen:

- Persönlichen Computer, Laptop, Tablet oder Mobiletelefon, welche mit dem Internet verbunden sind und Video streamen können.
- Einen oder mehrere Server auf welchen der ZAP Proxy, der Video Stream Analyser und der Splunk Server laufen. Der ZAP Proxy muss von den Teilnehmern ansprechbar und mit dem Internet verbunden sein.
- Splunk Light Version für die Teilnehmer zur Installation (ausser die Teilnehmer erstellen einen eigenen Account bei Splunk und downloaden die freie Light Version anschliessend selbst).
- Das Zertifikat des ZAP Proxys für die Teilnehmer.
- Verbindungsinformationen für die Teilnehmer (z.B. die IP-Adresse des ZAP Proxy).
- Gerät für motivierte Teilnehmer für die Vertiefung zuhause (siehe Use Case 02).



## 6 Auswertungen und Ergebnisse

### 6.1 Auswertungen mit Splunk

Es überrascht nicht, dass beim Vergleich der manuellen Analyse und der automatisierten Daten Erfassung die Resultate sehr ähnlich ausfallen. Die gemessenen Auflösungen und Bitraten sind gleich. Wobei die Bitraten für YouTube nicht automatisch erfasst werden kann.

Obwohl in beiden Fällen der Apposite Linktropy 5500 zum Einsatz kam, wird die Datenübertragungsrate mit dem Video Stream Analyser nicht erfasst, was eine Auswertung nach Übertragungsrate unmöglich macht. Aus diesem Grund können viele der erstellten Grafiken nicht direkt miteinander verglichen werden.

Der Video Stream Analyser erfasst zusätzlich die Abfragezeit und Grösse von allen Segmenten. Dadurch lässt sich die effektive Bitrate der Video Streams approximieren und mit der von den Providern vorgegebene Angaben vergleichen. Da es bei jedem Segment zu kurzen Verzögerungen kommen kann, wird bei der Analyse die Zeitspanne und berechnete Bitrate über zehn verschiedene Segment getätigt.

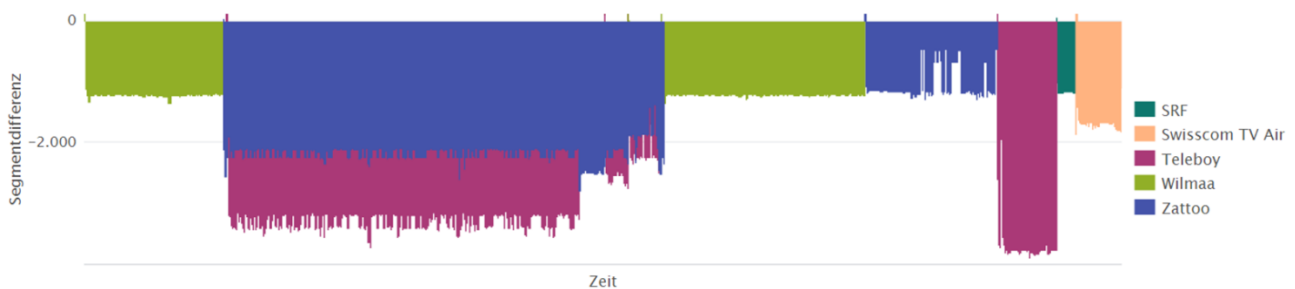


Abbildung 40: Differenz zwischen angegebener und effektiver Bitrate

Abbildung 40 zeigt die Differenz zwischen der approximiert effektiven Bitrate und der vom Portal vorgegebenen Bitrate. Die grossen Sprünge am Anfang eines jeweiligen Abschnitts stammen zum einen davon, dass beim Start eines Streams oft gleich mehrere Segmente innert kürzester Zeit geladen werden, zum anderen, da die Daten für die Grafik über die Differenz der Abfragezeit von zehn Segmenten berechnet wird. Wenn der Stream aus irgendeinem Grund pausiert wird und in dieser Zeit keine neuen Segmente geladen werden, so kommt es zu einer erhöhten Zeitspanne, welche dann in die nächsten zehn Datenzeilen mitgenommen wird. Durch einschränken des Zeitbereichs in denen es zu einem kontinuierlichen Stream gekommen ist, ist es möglich solche Probleme zu vermeiden.

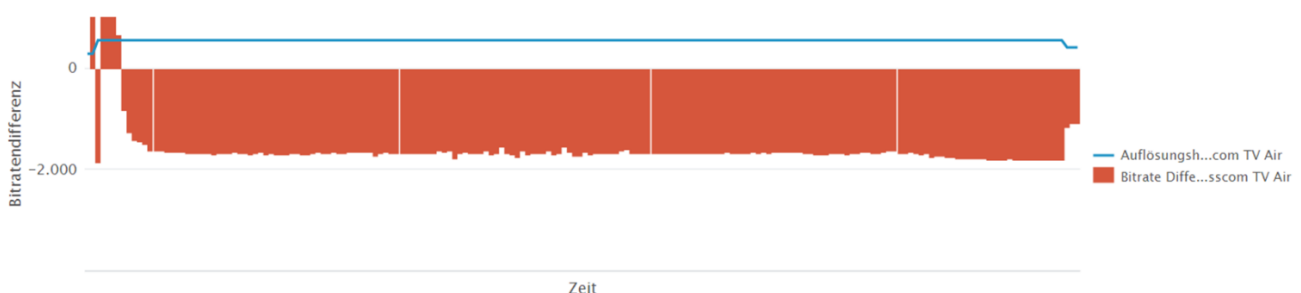


Abbildung 41: Bitratendifferenz für Swisscom TV Air

Interessant ist, dass nach einem erwarteten unpräzisen Start die Differenz sehr stabil wird. Es gibt jedoch bei jedem Provider eine tiefere Bitrate als die Angegebene. Eine Begründung für die generell tiefere Bitrate kann mit den analysierten Daten nicht gegeben werden. Spekulativ gesehen könnten die angegebenen Bitraten obere Limiten sein, während die berechneten Daten mit einer durchschnittlichen Bitrate operieren. Während es in einem Segment zu Bitratenspitzen kommen kann, so liegt der Durchschnitt unterhalb des Maximums.

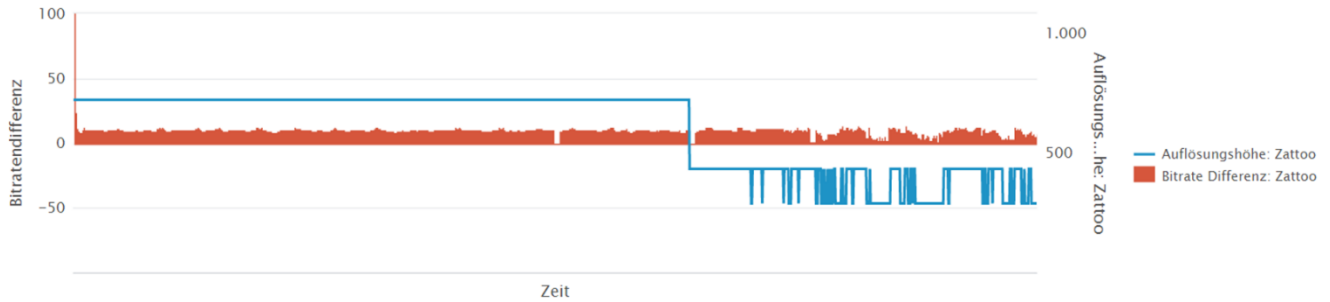


Abbildung 42: Bitratendifferenz in Prozent

Mit einer prozentualen Betrachtung der Bitratendifferenz wie in Abbildung 42, wird ersichtlich, dass der erwähnte Abstand prozentual gesehen immer etwa gleich gross bleibt.

## 6.2 Videostream Provider

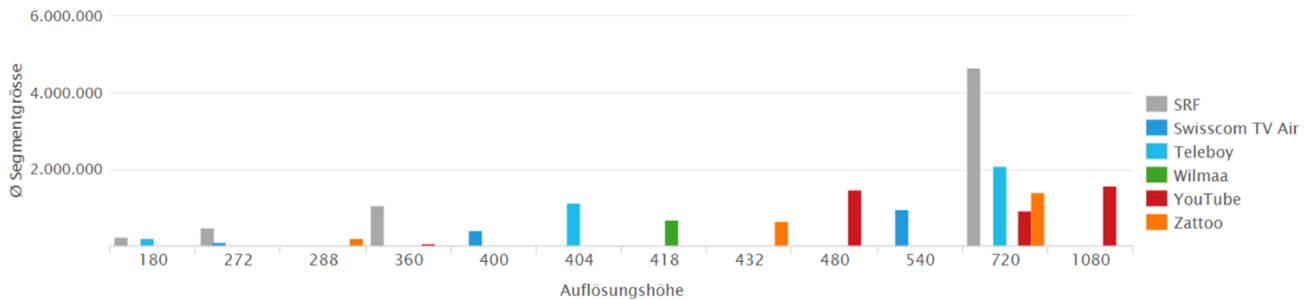


Abbildung 43: Durchschnittliche Segmentgrösse nach Auflösungshöhe

Da die Segmentgrösse linear abhängig ist zur Bitrate, wird in Abbildung 43 die durchschnittliche Segmentgrösse gegenüber der Auflösungshöhe gestellt. Dies hat den Vorteil, dass YouTube ebenfalls mit in die Grafik genommen wird, obwohl es für YouTube bis jetzt noch nicht möglich ist, die Bitrate automatisch zu erfassen. Das SRF Videoportal verwendet bei den angebotenen Replay Videos sehr grosse Segmente, weshalb es in Abbildung 43 stark auffällt.

### 6.2.1 Teleboy

Teleboy fällt stark auf, zum einen wurden Daten über eine grössere Vielzahl von Auflösungshöhen gesammelt, zum anderen verwendet Teleboy im Vergleich die grössten Segmente und somit bietet auch die höchste Auflösung, wie in Abbildung 44 ersichtlich wird.

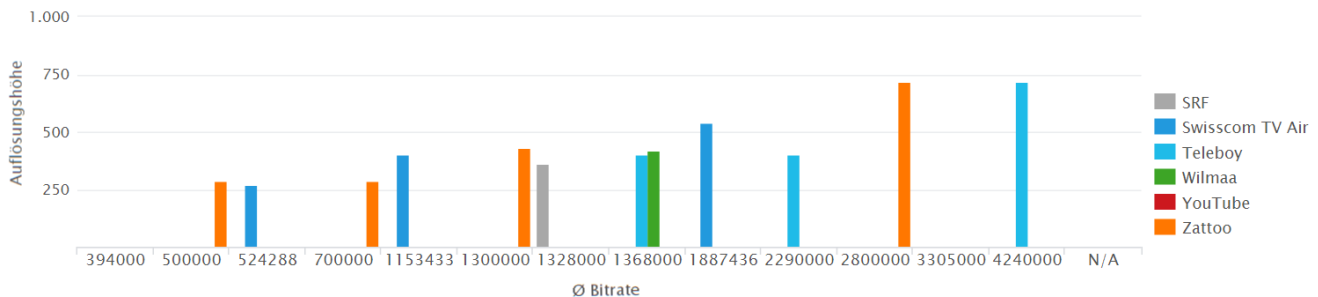


Abbildung 44: Durchschnittliche Bitrate nach Auflösungshöhe

Ebenso fällt in Abbildung 44 auf, dass Teleboy als auch Zattoo für dieselbe Auflösungshöhe zwei verschiedene Bitraten anbietet. Für den Kunden bedeutet dies, dass bei einer schlechteren Internetverbindung immer noch gleich viele Pixel angezeigt werden, während die benötigte Übertragungsrate abnimmt. Bei gleichbleibender Auflösung aber geringerer Bitrate, müssen die Videoinformationen stärker komprimiert werden, was sich durch ausgeprägtere Artefakte im Video zeigt. Obwohl dieser Komprimierungsfaktor nicht als Parameter erfasst wird, hat dies einen geringen Effekt auf die Auswertung, denn die Beurteilung ob stärker komprimierte Videos von den Kunden als schlechter wahrgenommen werden, ist eine stark subjektive Angelegenheit. Diese Verbindung zwischen subjektiver Aussage und den automatisch erfassten Daten wurde in dieser Arbeit nicht gemacht, würde sich aber durch Benutzertests relativ gut erheben lassen.

Negativ zu bemängeln gibt es nur die Schaltung von Werbung, ist aber für das gratis Angebot nicht überraschend. Sehr mühsam wird die Werbung jedoch bei tiefen Übertragungsraten, da das Laden der Werbung eine lange Zeit dauert. Bleibt der Videostream hängen, so muss die Webseite neugeladen werden und erneut gibt es bis zu 30 Sekunden Werbung zu betrachten.

### 6.2.2 Zattoo

Zattoo passt in eine ähnliche Kategorie wie Teleboy. Die automatische Skalierung funktioniert einwandfrei und die Bitrate ist im Vergleich gut gewählt. Im Vergleich zu Teleboy ist die Bitrate bei Zattoo für die HD Auflösung um einiges kleiner.

Bevor die eigentliche Zattoo Webseite lädt, wird bereits ein Werbespot angezeigt. Danach werden, wie bei anderen Providern, beim Öffnen eines Kanals erneut zwei Werbespots gezeigt. Nach einigen Sekunden können die Spots meistens übersprungen werden, aber da es verschiedene Varianten in der Darstellung gibt, einmal unten neben dem Countdown, dann wieder auf dem Video mit einem Pfeil, kann es schnell mal zu Verwirrung führen. So klickt man aus Versehen auf den Werbespot oder schaut sich den gesamten Spot an.

### 6.2.3 YouTube

YouTube zeichnet ein inkonsistentes Bild ab, was nicht überraschend ist. Andere Videostreaming Provider konzentrieren sich auf das Übertragen von Fernsehsender. YouTube jedoch bietet Videos aller Art an. Da die Videos von den Benutzern selbst erstellt und hochgeladen werden, kann es auch grosse Unterschiede in der Bitrate und den verfügbaren Auflösungen geben. Falls zum Beispiel jemand einen Livestream mit Full HD Auflösung startet, aber mit sehr tiefer Bitrate, so kann YouTube selbst nichts tun, um die Qualität zu verbessern und wird dementsprechend die tiefe Bitrate weitergeben.

YouTube bietet als einziger Provider Auflösungen von 256x144, über Full HD bis zu 8K (7680x4320) an. Der Benutzer muss lediglich die höchste Stufe hochladen, welche YouTube neu encodiert und davon automatisch tiefere Auflösungsstufen generiert.

Die Skalierung bei YouTube ist eine der besten und auch bei geringster Datenübertragungsrate ist es immer noch möglich Videos zu schauen.

Einen konkreten Vergleich zu anderen Providern ist nicht möglich, da der Inhalt von den Videos auf YouTube von den Benutzern und nicht Fernsehsendern erstellt wird und YouTube weltweit unter Alphabet Inc. (vormalig Google Inc.) operiert, somit auch weit mehr Möglichkeiten hat und andere Use Cases umsetzen müssen.

### 6.2.4 Swisscom TV Air

Der Name Swisscom TV Air kann schnell mal zu Verwirrung führen, denn die anderen Swisscom Pakete fangen ebenfalls mit Swisscom TV an. Swisscom TV Air unterscheidet sich darin, dass es von überall ohne zusätzliches Gerät verwendet werden kann [24].

	Swisscom TV L Kunden	Swisscom TV M Kunden	Swisscom TV S Kunden	inOne mobile XL – XS Kunden*	TV Air free Kunden
TV-Sender	300	270	100	200	200
Replay	7 Tage	7 Tage	–	30h	30h
Aufnahme	2400 h	240 h	–	60h	60h
Aufnahme offline ansehen	✓	✓	✓	✓	–
Filme mieten	PC	PC	PC	PC	–
Live Sport Events	✓	✓	✓	✓	–
Teleclub Play	✓	✓	✓	✓	–
HD-Qualität	✓	✓	✓	✓	–
Fernbedienung zur TV-Box zuhause	✓	✓	✓	–	–
Persönliche Senderreihenfolge	✓	✓	✓	✓	✓
Programm-Tipps	✓	✓	✓	✓	✓
Live Pause	✓	✓	✓	✓	✓
Werbefrei	✓	✓	✓	✓	–
Gratis Streaming**	✓	✓	✓	✓	–
Login	TV-Login	TV-Login	TV-Login	inOne mobile oder NATEL® infinity Login	Facebook-Login oder Mobile-Nr.
Swisscom Abo	Ja	Ja	Ja	Ja	Nein
Kosten	inkl.	inkl.	inkl.	inkl.*	0.–/Mt

Abbildung 45: Swisscom TV Angebote

Swisscom TV Air bietet keine HD Auflösung an und reiht sich somit hinter Teleboy und Zattoo ein. Aus der Abbildung 45 wird klar, dass bessere Auflösungen für die kommerziellen Abonnements vorbehalten sind.

Angenehmerweise zeigt Swisscom TV Air jeweils nur einen Werbespot bevor der Videostream beginnt.

### 6.2.5 Wilmaa

---

Wilmaa schneidet mit einer fixen Auflösung und einer fixen Bitrate im Vergleich schlecht ab. Bei tiefer Übertragungsrate kommt es zu vielen Unterbrüchen und bei über 1.5 Mbit/s skaliert die Auflösung und Bitrate und somit die gesamte Videoqualität nicht weiter nach oben.

Etwas positiver fällt hier die Anwendung von Werbung auf, mit nur einem Clip vor dem Start. Für die weitere Monetisierung werden zusätzlich noch Werbungskanäle verwendet, welche meist standardmässig ausgewählt werden.

### 6.2.6 SRF

---

Das SRF Videoportal entpuppt sich wiederum als spezieller Fall, denn es werden jeweils nur die drei Schweizer Fernsehkanäle als Live Videos angeboten und der Rest sind Replay Videos. Während die Live Videos nur eine Auflösung von Maximal 640x360 bieten, so gehen Replay Videos bis zu HD Auflösungen.

Beim Laden von Replay Videos werden sehr grosse Segmentgrößen geladen wie auf Abbildung 43 ersichtlich ist. Dies führt dazu, dass mehr Zeit benötigt wird, um das nächste Segment herunterzuladen, was wiederum Probleme beim Start eines Videos geben kann, da noch nicht genügend Videomaterial im Buffer vorhanden ist.

## 7 Schlussfolgerung

---

### 7.1 Zusammenfassung

---

Im Rahmen dieser Bachelorarbeit wurden die sieben Videoportale YouTube, Netflix, SRF, Zattoo, Swisscom TV Air, Teleboy und Wilmaa analysiert. Zur Simulation von verschiedenen und verändernden Internetverbindungsbedingungen wurde ein Apposite Linktropy 5500 eingesetzt.

Die Analyse gab Einblick in die Funktionsweise von Videostreaming Portalen und den involvierten Technologien, wie zum Beispiel die Streaming Protokolle HLS und DASH. Desweiteren wurde eine Liste von teilweise gemeinsamen Parametern, wie zum Beispiel Bitrate, Auflösung oder Segmentgrösse, erstellt.

Basierend auf diesem Wissen wurde ein Video Stream Analyser entwickelt, welcher mit dem OWASP ZAP Proxy den Datenverkehr abfängt und nach aktiven Videostreams durchsucht. Die gefundenen Streams werden getrackt und die definierten Parameter als CSV abgespeichert.

Die CSV Dateien werden an Splunk weitergeleitet. Splunk verfügt über eine eigene Suchsprache und bietet verschiedenste Visualisierungsmöglichkeiten an. Es wurden zusätzlich zwei Beispiel Dashboards erarbeitet, um mögliche Analysen mit Splunk aufzuzeigen.

Unter den Gratisangeboten von Fernseh-Videoportalen positioniert sich Teleboy an erster Stelle. Mit der besten Bitrate und einer HD Auflösung schlägt es die anderen Portale, während es sich beim Anzeigen von Werbung in etwa gleich verhält.

Für Unterhaltung jeglicher Art bleibt YouTube ungeschlagen, durch hohe Auflösungen, gut optimierte Videostreams und endlos viele Auswahlmöglichkeiten.

Das SRF Portal bietet insbesondere von SRF produzierte Videos an, welche zum grössten Teil nur dort verfügbar sind. Die Wiedergabe könnte durch kleinere Segmente noch etwas optimiert werden.

### 7.2 Ausblick

---

Der Video Stream Analyser wurde nur unter optimalen Bedingungen getestet und somit ist es unklar wie viele Bugs nicht entdeckt wurden. Mit weiterem Testen und Anpassen sollte aus der Anwendung ein stabiles Analysetool werden.

Das Testen auf dem fitPC zeigte auf, dass der ZAP Proxy unter Belastung einen leistungsfähigen Prozessor braucht. Eine möglichst portable Lösung muss noch gefunden und getestet werden.

Der erarbeitete Splunk Kurs sollte als gute Grundlage dienen, muss aber sicherlich noch so angepasst werden, dass die Ideen und Ziele des Kursleiters Einfluss finden. Zusätzlich ist es interessant zu sehen, was für weitere Visualisierungen und Analyse andere Gruppen finden können.

Diese Bachelorarbeit lässt noch viele Möglichkeiten für zukünftige Arbeiten offen. Es wäre zum Beispiel interessant, Benutzertests durchzuführen um die subjektive Bewertung von verschiedenen Videoqualitäten in Erfahrung zu bringen, oder aber den Fokus auf einen ganz anderen Bereich wie zum Beispiel Online Spiele zu richten.

## 8 Anhang

---

### 8.1 Verzeichnisse

---

#### 8.1.1 Literatur- und Quellenverzeichnis

---

- [1] Cnlab Information Technology Research AG, „Page Loading Time,“ 2016.
- [2] A. B. Watson, „Temporal Sensitivity,“ NASA Ames Research Center, Moffett Field, California, 1986.
- [3] Linus Media Group, „YouTube,“ Techquickie, 8 August 2014. [Online]. Available: <https://www.youtube.com/watch?v=qbGQBT2Vwvc>. [Zugriff am 8 June 2017].
- [4] P. D. I. K. Lichtscheidl, „Intraframe und Interframe-Kompression,“ Cell Imaging und Ultrastrukturforschung - Universität Wien, [Online]. Available: <https://www.univie.ac.at/video/grundlagen/intraframe.htm>. [Zugriff am 8 Juni 2017].
- [5] R. S. Sabeenian und S. Ponlatha, „Comparison of Video Compression Standards,“ *International Journal of Computer and Electrical Engineering*, Bd. 5, Nr. 6, p. 549, 2013.
- [6] Encoding.com, „Global Media Format Report,“ Various sources (Encoding.com), 2016.
- [7] „How MPEG-TS works,“ VideoHelp Forum, 3 Juni 2009. [Online]. Available: <https://forum.videohelp.com/threads/306126-HFS10-AVCHD-how-to-maintain-quality#post1881643>. [Zugriff am 8 Juni 2017].
- [8] Digital Video Broadcasting Project, „TelCo Group,“ Juni 2010. [Online]. Available: [https://telcogroup.ru/files/materials-pdf/DVB\\_standards/DVB-C/a138\\_DVB-C2.pdf](https://telcogroup.ru/files/materials-pdf/DVB_standards/DVB-C/a138_DVB-C2.pdf). [Zugriff am 8 Juni 2017].
- [9] RFC, „RFC,“ [Online]. Available: <https://tools.ietf.org/html/rfc2616#section-1.4>. [Zugriff am 24 März 2017].
- [10] Apple Inc., „HTTP Live Streaming,“ [Online]. Available: <https://developer.apple.com/streaming/>. [Zugriff am 8 Juni 2017].
- [11] ISO/IEC, 23009-1 - Dynamic adaptive streaming over HTTP (DASH), Switzerland: ISO/IEC, 2014.
- [12] Black Duck Software, Inc., „OWASP Portfolio Projects,“ May 2017. [Online]. Available: [https://www.openhub.net/orgs/OWASP?view=portfolio\\_projects](https://www.openhub.net/orgs/OWASP?view=portfolio_projects). [Zugriff am 15 May 2017].
- [13] Splunk, Inc., „Splunk,“ May 2017. [Online]. Available: <https://www.splunk.com>. [Zugriff am 15 May 2017].
- [14] Leichtman Research, „Leichtman Research,“ [Online]. Available: <http://www.leichtmanresearch.com/press/030617release.html>. [Zugriff am 24 03 2017].
- [15] Gold Media, „Pay-VOD to become billion-euro market in Germany,“ Gold Media, Berlin, 2016.

- [16] Cnlab Information Technology Research AG, „DNS Ranking Schweiz für Video Dienste,“ Rapperswil, 2017.
- [17] Digital TV Research, „Statista - The Statistics Portal,“ Digital TV Research, September 2014. [Online]. Available: <https://www.statista.com/statistics/324568/penetration-rate-netflix-country/>. [Zugriff am 24 März 2017].
- [18] Valve, „Steam Hardware & Software Statistics,“ February 2017. [Online]. Available: <http://store.steampowered.com/hwsurvey?platform=combined>. [Zugriff am 24 März 2017].
- [19] StatCounter, „StatCounter,“ 2016. [Online]. Available: <http://gs.statcounter.com/screen-resolution-stats/desktop/worldwide#monthly-201701-201702-bar>. [Zugriff am 24 März 2017].
- [20] Computerphile, „YouTube,“ Computerphile, 16 November 2013. [Online]. Available: <https://www.youtube.com/watch?v=OqQk7kLuaK4>. [Zugriff am 8 Juni 2017].
- [21] YouTube, „YouTube Answer,“ [Online]. Available: <https://support.google.com/youtube/answer/2853702>. [Zugriff am 24 März 2017].
- [22] Netflix, „Netflix,“ [Online]. Available: <http://techblog.netflix.com/2015/12/per-title-encode-optimization.html>. [Zugriff am 24 März 2017].
- [23] Splunk, Inc., „Splunk Light,“ May 2017. [Online]. Available: [https://www.splunk.com/en\\_us/products/splunk-light.html](https://www.splunk.com/en_us/products/splunk-light.html). [Zugriff am 15 May 2017].
- [24] Swisscom, „TV Air Swisscom TV auf PC, Tablet und Smartphone,“ Swisscom, [Online]. Available: <https://www.swisscom.ch/de/privatkunden/internet-fernsehen-festnetz/digital-tv/tv-air.html>. [Zugriff am 14 Juni 2017].
- [25] MarketingCharts, „Statista,“ Hitwise, November 2016. [Online]. Available: <https://www.statista.com/statistics/266201/us-market-share-of-leading-internet-video-portals/>. [Zugriff am 24 März 2017].



### 8.1.2 Glossar und Abkürzungsverzeichnis

API	Application Programming Interface
Auflösung	Anzahl Pixels in einem Frame, meist durch Höhe und Breite angegeben
Bitrate	Anzahl verarbeiteter Bits pro Sekunde
Bildpunkt	Kleinster, einzelner Punkt eines Frames
Bildwiederholungsrate	Anzahl angezeigter Frames pro Sekunde
Buffering	Zwischenspeicherung von Videodaten
Client	Lokaler Computer der auf einen Server zugreift
Codec	Algorithmenpaar das Daten digital kodiert und decodiert, siehe Abschnitt 2.1.2
Containerformat	Dateiformate welche Video- und Audiostreams, sowie weitere Daten enthalten können, siehe Abschnitt 2.1.3
DASH	Dynamic Adaptive Streaming over HTTP, siehe Abschnitt 2.2.6.2
Decoder	Anwendung des Codecs um ein Videostream in Frames für die Anzeige zu zerlegen
Dropped Frames	Frame das nicht gerendert wurde
Encoder	Anwendung des Codecs um Frames in einen Videostream umzuwandeln
FPS	Frames pro Sekunde
Frame	Einzelnes Bild eines Videos
Framerate	Siehe Bildwiederholungsrate
Full HD	Auflösung von 1920x1080 Pixels
HD	Auflösung von 1280x720 Pixels
HLS	HTTP Live Streaming, siehe Abschnitt 2.2.6.1
HTTP	Hypertext Transfer Protocol
Live Video	Zeitnah aufgezeichnetes und gestreamtes Video, siehe Abschnitt 2.2.3
Lossless	Verlustfrei
Lossy	Verlustbehaftet
Paketverlust	Wenn Netzwerkpakete von Server zum Client verloren gehen
Pixel	Siehe Bildpunkt
QUIC	Quick UDP Internet Connections, siehe Abschnitt 2.2.5.3
Recoverytime	Zeit zwischen Unterbrechung und Fortsetzung eines Videostreams
Regex	Regular Expression
Replay Video	Videos welche nicht Zeit gebunden gestreamt werden müssen, siehe Abschnitt 2.2.3
Segment	Abgeschlossener Videoausschnitt der in sich abgespielt werden kann,

	siehe Abschnitt 2.2.1
Server	Externer Computer auf welche Clients zugreifen können
Set-Top-Box	Übersetzt das Videosignal vom Provider für die Darstellung am Fernseher
SPL	Search Processing Language, Splunks eigene Abfragesprache
STB	Siehe Set-Top-Box
Stockende Unterbrechung	Videostream stoppt und fährt an derselben Stelle weiter, siehe Abschnitt 2.2.4.2
Stream	Gleichzeitiges Übertragen und Empfangen von Video- und Audiodaten, siehe Abschnitt 2.2.1
TCP	Transmission Control Protocol, siehe Abschnitt 2.2.5.1
UDP	User Datagram Protocol, siehe Abschnitt 2.2.5.2
Unterbrechung	Videostream stoppt ohne Benutzereinwirkung
Überspringende Unterbrechung	Videostream stoppt und fährt an einer neuen Stelle weiter, siehe Abschnitt 2.2.4.1

### 8.1.3 Abbildungen

Abbildung 1: Seitenladezeit bei verschiedenen Anbietern.....	12
Abbildung 2: Allgemeiner Prozess von Videos.....	14
Abbildung 3: Komprimierung mit I-, B- und P-Frames.....	14
Abbildung 4: Übersicht Videostreaming.....	16
Abbildung 5: Grober Aufbau eines Segments.....	17
Abbildung 6: Buffergrösse im YouTube Player.....	18
Abbildung 7: Apposite Linktropy 5500.....	21
Abbildung 8: Setup mit Apposite Linktropy.....	21
Abbildung 9: Apposite Einstiegsseite.....	22
Abbildung 10: Der Apposite Linktropy Scheduler.....	23
Abbildung 11: Verwendung von ZAP.....	25
Abbildung 12: Funktionsweise von Splunk.....	26
Abbildung 13: Führende Multimedia Webseiten in den USA (2016).....	28
Abbildung 14: Videoportale in Deutschland (2016).....	29
Abbildung 15: DNS Abfragen für Videoportale bei Swisscom via cnlab.....	30
Abbildung 16: Prognose zum Verbreitungsgrad von Netflix in der Schweiz und in Deutschland.....	30
Abbildung 17: Ausgewählte Videoportale.....	31
Abbildung 18: Experiment Aufbau.....	32
Abbildung 19: Auflösungen von Live Videostreams.....	35
Abbildung 20: Bitraten von Live Videostreams.....	36
Abbildung 21: Bitraten Verhältnis von Live Videostreams.....	37
Abbildung 22: Auflösung im Vergleich zur Bitrate für Live Videostreams.....	38
Abbildung 23: Auflösung von Live Videostreams.....	39

---

Abbildung 24: Auflösungen von Replay Videostreams .....	39
Abbildung 25: Bitraten von Live Videostreams .....	40
Abbildung 26: Auflösungsvergleiche Normal vs. Paketverlust .....	41
Abbildung 27: Bitraten von Replay Videostreams .....	41
Abbildung 28: Biraten Verhältnis für Replay Videostreams .....	42
Abbildung 29: Infrastruktur des Video Stream Analysers .....	43
Abbildung 30: Struktur des Video Stream Analysers .....	44
Abbildung 31: Interner Ablauf des Video Stream Analysers .....	45
Abbildung 32: Splunk Dateneingabe .....	47
Abbildung 33: Splunk Überwachen .....	48
Abbildung 34: Splunk Suche .....	49
Abbildung 35: Splunk Ereigniseinsicht .....	49
Abbildung 36: Splunk Suchbefehle .....	50
Abbildung 37: Splunk Dashboard XML .....	51
Abbildung 38: Splunk Kurs Setup UC01 .....	53
Abbildung 39: Splunk Kurs Setup UC02 .....	54
Abbildung 40: Differenz zwischen angegebener und effektiver Bitrate .....	56
Abbildung 41: Bitratendifferenz für Swisscom TV Air .....	56
Abbildung 42: Bitratendifferenz in Prozent .....	57
Abbildung 43: Durchschnittliche Segmentgrösse nach Auflösungshöhe .....	57
Abbildung 44: Durchschnittliche Bitrate nach Auflösungshöhe .....	58
Abbildung 45: Swisscom TV Angebote .....	59
Abbildung 46: Projektiterationen .....	67
Abbildung 47: Theoretischer Aufwand vs. IST-Aufwand .....	68
Abbildung 48: IST-Aufwand jedes Entwicklers .....	69
Abbildung 49: Aufwandsanteil pro Iteration .....	70

### 8.1.4 Tabellen

---

Tabelle 1: Am häufigsten angetroffene Auflösungen .....	34
Tabelle 2: Videostream Merkmale .....	46
Tabelle 3: Aktoren .....	53

### 8.1.5 Formeln

---

Formel 1: Berechnung nicht komprimierter, maximaler Bitrate .....	36
---	----

## 8.2 Projektmanagement

### 8.2.1 Projektplan

SW1	SW2	SW3	SW4	SW5	SW6	SW7	SW8	SW9	SW10	SW11	SW12	SW13	SW14	SW15	EW1	
24.02.17	03.03.17	10.03.17	17.03.17	24.03.17	31.03.17	07.04.17	14.04.17	21.04.17	28.04.17	05.05.17	12.05.17	19.05.17	26.05.17	02.06.17	09.06.17	
Inception		Elaboration 1			Elaboration 2			Construction 1			Construction 2			Transition 1		Transition 2
M1 Aufgabenstellung		M2 verschiedener Methoden <sup>1)</sup>			M3 Erfassung und Analyse <sup>2)</sup>			M4 zur Erfassung des Benutzerverhaltens			M5 eines Splunk-Forwarder-Prototypen			M6 Dokumentationsüberarbeitung		M7 Arbeitsabgabe

<sup>1)</sup> Abgabe Analysebericht

<sup>2)</sup> inkl. Zwischenpräsentation

SW1 = Semesterwoche 1

EW1 = Extrawoche 1

Datum = jeweils Freitag der Woche

M1 = Meilenstein 1

Abbildung 46: Projektiterationen

Die Zeiteinteilung der Arbeit wurde auf die einzelnen Iterationen abgestimmt. Mit einem agilen Projektplan wurde die Einarbeitung aller Beteiligten in das Streaming-Umfeld berücksichtigt und die Möglichkeit der Verschiebung einzelner Prioritäten gewährleistet.

#### 8.2.1.1 Inception

Während der Inception Phase wird die Aufgabenstellung genauer analysiert und das Verständnis im Team sichergestellt. Im Weiteren richtet sich das Team im BA-Zimmer der HSR ein und installiert das Setup gemäss dem Arbeitsverständnis.

#### 8.2.1.2 Elaboration

In den Elaboration Phasen werden verschiedene Methoden zur Problemlösung evaluiert. Ein Grossteil der Zeit steht für die Analyse des Streaming-Umfelds zur Verfügung, in welchem die verschiedenen Protokolle und Varianten der Streaming-Möglichkeiten einzelner Videoportale analysiert werden. Anschliessend müssen zusätzliche Parameter zur Datenübertragungsrate gefunden werden, welche die Benutzererfahrung beeinflussen können. Schlussendlich wird eine manuelle Erfassung und Auswertung der definierten Parameter durchgeführt und besprochen. Zusätzlich kann die Aufgabenstellung neuen Erkenntnissen angepasst werden.

Das primäre Ziel der Elaboration Phasen ist ein professionelles Verständnis der Streaming-Methoden der einzelnen Videoprovider.

### 8.2.1.3 Construction

Die Construction Phasen sind insbesondere der Entwicklung eines Video Stream Analyser gewidmet. Die erarbeiteten Optionen sollen nach Effizienz in der Erarbeitung und Ausführung beurteilt werden und das Team wird sich in die benötigten Applikationen der ausgewählten Option einarbeiten. Ein weiterer wichtiger Punkt der Construction Phasen ist die Splunk Software und die Auswertungsmöglichkeiten der Parameter in Verbindung mit dem zu entwickelnden Video Stream Analyser.

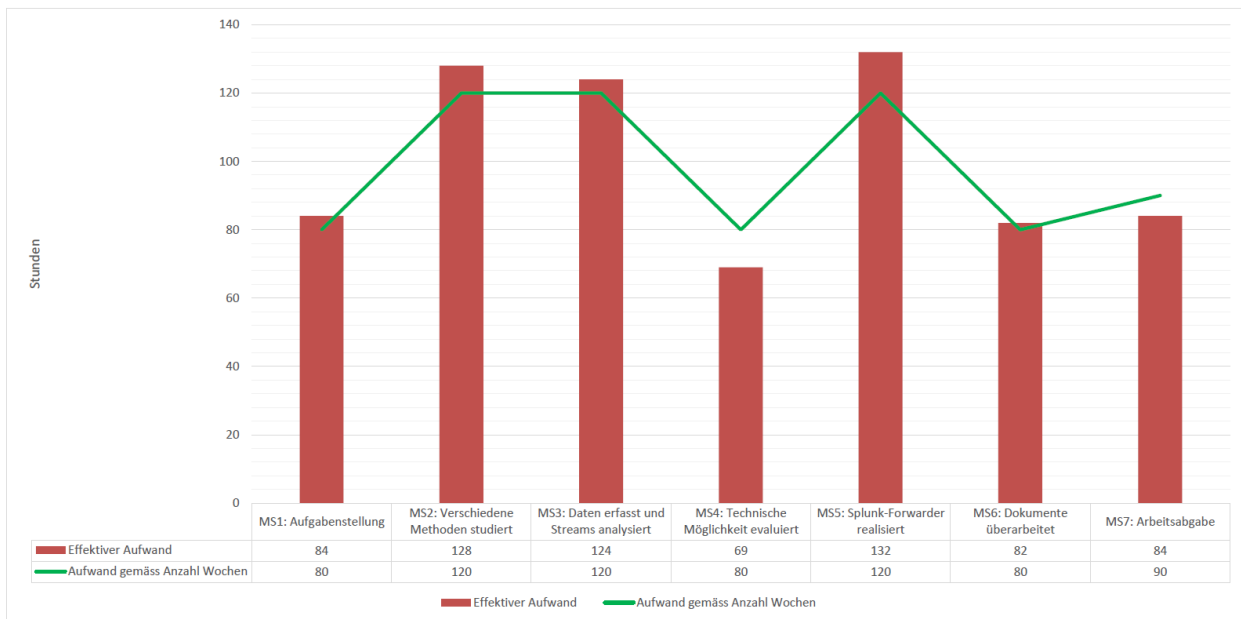
Das primäre Ziel der Construction Phasen ist ein funktionierender Video Stream Analyser und dokumentierte Ideen für einen Splunk Kurs.

### 8.2.1.4 Transition

Das Hauptaugenmerk der Transition Phasen ist die Überarbeitung und Vervollständigung aller Dokumente, insbesondere der Dokumentation der Bachelorarbeit.

## 8.2.2 Zeitauswertung

### Theoretischer Aufwand vs. IST-Aufwand



**Abbildung 47: Theoretischer Aufwand vs. IST-Aufwand**

In diesem Diagramm ist erkennbar wie der effektive Aufwand der Arbeit sich, jeweils in Arbeitsstunden, über die einzelnen Meilensteine verteilt. Für die Berechnung des effektiven Aufwands wurden die erbrachten Stunden beider Entwickler für die jeweiligen, im Meilenstein enthaltenen, Wochen summiert.

Die grüne Linie zeigt den theoretischen Aufwand pro Meilenstein an. Für diesen wird der zu erwartende Arbeitsaufwand durch die Anzahl verfügbarer Wochen dividiert, um einen theoretischen Aufwand pro Woche zu ermitteln. Hier geht man von der Annahme aus, es würde jede Woche gleich viel Zeit in das Projekt investiert werden. Für das Diagramm wird anschliessend die Anzahl Wochen pro Meilenstein mit dem errechneten Arbeitsaufwand pro Woche multipliziert.

Dadurch wird im Diagramm übersichtlich angezeigt, dass in den Meilensteinen 2, 3 und 5 mehr Zeit als durchschnittlich notwendig investiert wurde, wohingegen im Meilenstein 4 weniger Zeit eingesetzt wurde.

### IST-Aufwand jedes Entwicklers

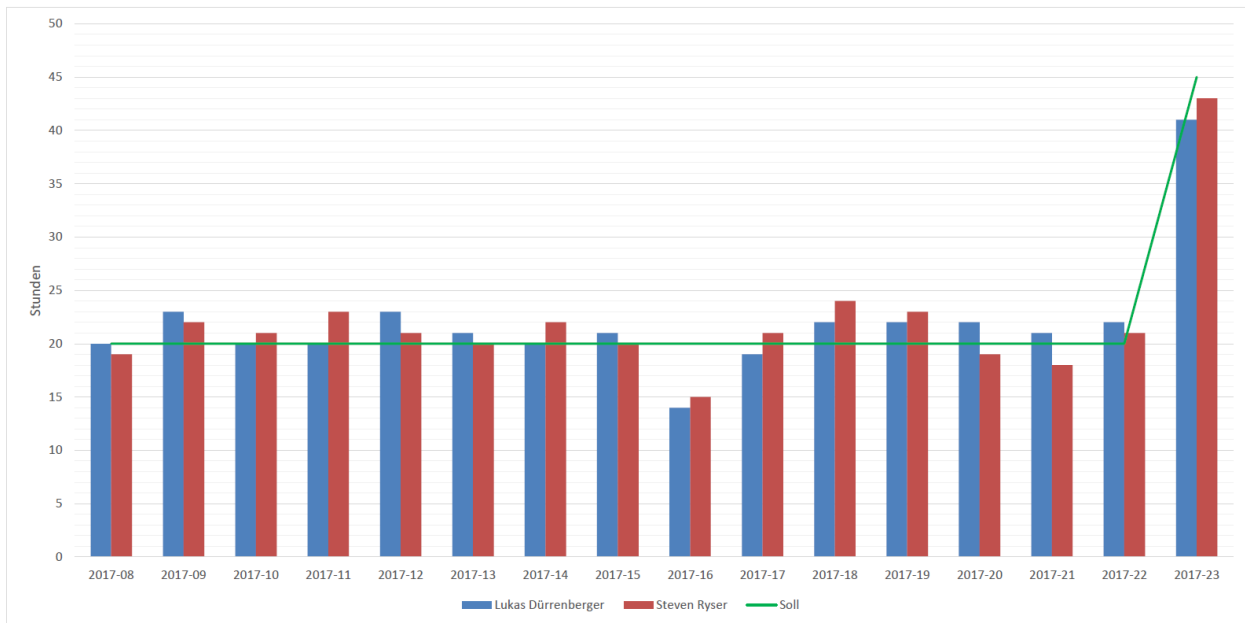


Abbildung 48: IST-Aufwand jedes Entwicklers

Dieses Diagramm stellt die Verteilung des individuellen Aufwands über die 16 Wochen des Projekts anschaulich dar. Man erkennt an der Fluktuation der jeweiligen Wochen, wie unterschiedlich diese Wochen zum Teil waren und wann mehr oder weniger am Projekt gearbeitet werden konnte. Leicht erkennbar ist zum Beispiel die Tatsache, dass in der 16. Kalenderwoche weniger am Projekt gearbeitet wurde, was sich mit dem verlängerten Wochenende über Ostern und einem Mehraufwand für andere zwischensemestrigere Modulabgaben erklären lässt.

Die grüne Linie zeigt den theoretischen Aufwand pro Woche an. Um diesen zu ermitteln wird der zu erwartende Arbeitsaufwand durch die Anzahl verfügbarer Wochen im Projekt dividiert. Dabei wird von der Annahme ausgegangen, dass jede Woche gleich viel Zeit investiert werden würde, mit Ausnahme der letzten Woche, die vollständig der Arbeit am Projekt gewidmet ist.

Sichtbar wird durch das Diagramm auch, dass manchmal der eine Entwickler etwas mehr zu tun hatte und ein andermal der andere. Schlussendlich zeigt sich durch die verhältnismässige Gleichverteilung aber, dass beide Entwickler gleich viel zum Erfolg der Arbeit beigetragen haben, was sich in der guten Teamarbeit widerspiegelt.

## Aufwandsanteil pro Iteration

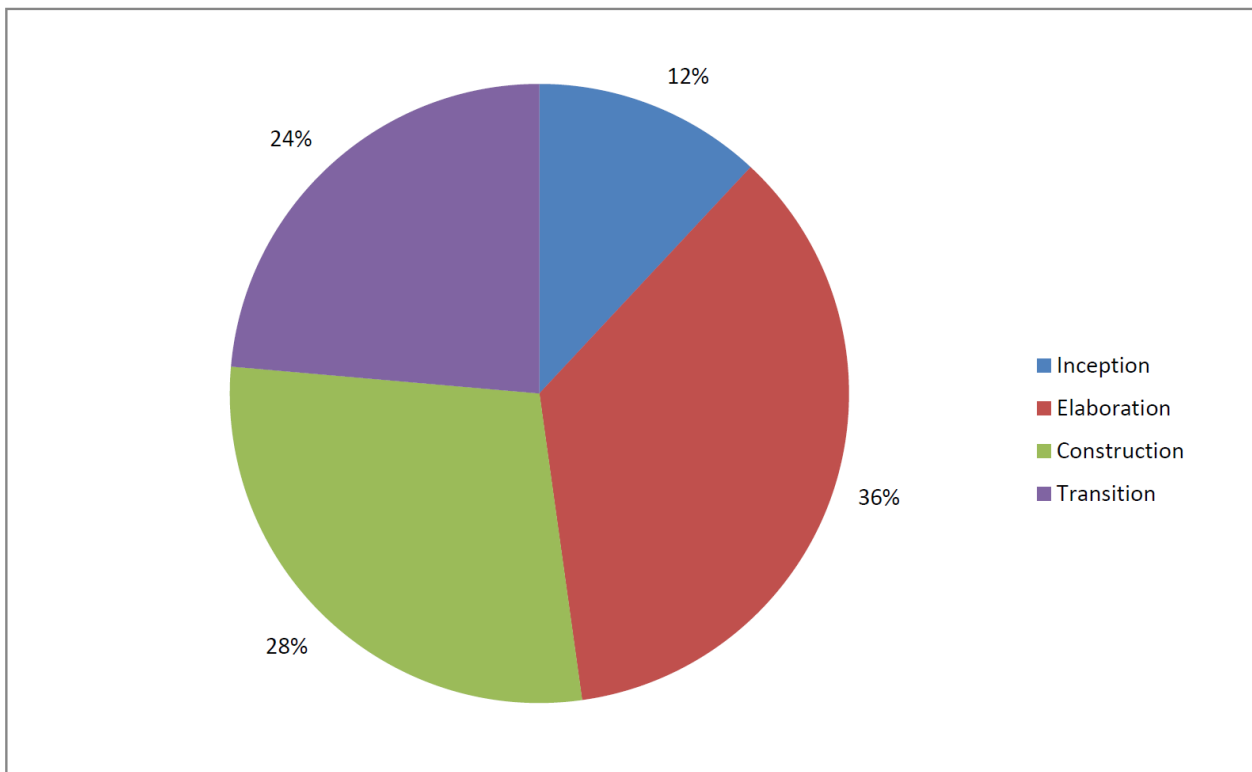


Abbildung 49: Aufwandsanteil pro Iteration

Hier ist die Verteilung des effektiven Aufwands pro Meilenstein im Verhältnis zum Gesamtaufwand erkennbar. Es wird schnell offenkundig, dass die Phasen Elaboration und Construction die meiste Zeit beanspruchten, was auch Sinn macht und darum auch im Projektplan so eingeplant wurde. Die Elaboration Phase hat verhältnismässig mit Abstand am meisten Zeit benötigt, was mit dem Einarbeiten in die Streaming-Umgebung und die Analyse der einzelnen Videoportale zusammenhängt.

---

## 8.3 Persönliche Berichte

---

### 8.3.1 Lukas Dürrenberger

---

Das Thema zu Internet Speed und das Kundenerlebnis hat mich schon längere Zeit interessiert, unter anderem weil meine Eltern etwas abgelegen wohnen und ich dort praktisch jedes Wochenende in irgendeiner Art und Weise ans Limit gerate. Dies war dann auch eine der stärkeren Motivationen, mich mit dem Thema noch genauer auseinanderzusetzen.

Die Aufgabenstellung war zu Beginn sehr offen formuliert, was mir noch so einige Fragen bereitete. Doch schnell durfte ich feststellen, dass, durch gute Gespräche mit Prof. Heinzmann und Eric Franke, wir uns dem Ziel Schritt für Schritt annähern konnten.

Das Arbeiten mit konkreten Daten gefiel mir sehr gut, insbesondere die verschiedenen Arten Daten zu Visualisierung. Ich hoffe, dass unsere Arbeit als gute Grundlage für den Splunk Kurs dient und dass noch weitere interessante Zusammenhänge gefunden werden können.

Ich hatte bis jetzt noch relativ wenig Erfahrung mit Python, aber wusste von einigen kleineren Experimenten, dass es sich sehr gut für solche Aufgaben eignet, denn es wir konnten sehr einfach bereits existierenden Code einbinden und mussten uns somit weder um die Anbindung zum ZAP, noch um das Parsen von den Videostream Manifests kümmern. Obwohl mir der Aufbau sehr gut gefällt, so würde ich im Nachhinein vielleicht einen etwas anderen Ablauf verwenden, der das Erkennen von bereits laufenden Streams ebenfalls erfasst.

### 8.3.2 Steven Ryser

---

Die Bachelorarbeit war eine sehr spannende, wenn auch anstrengende Zeit. Diese 15 Wochen waren schneller vorbei als ich anfangs dachte, aber in den stundenlangen Teilarbeiten habe ich viel gelernt. Das Thema des Projekts fand ich von Beginn an interessant und die Einarbeitung in die Streaming-Technologie war fordernd und lehrreich. Den Video Stream Analyser in Python zu programmieren war eine neue Erfahrung für mich, da ich noch nie etwas mit Python gemacht habe. Doch mit einigen Recherchen und gutem Teamwork formte sich rasch eine funktionierende Software. Zusätzlich dazu war es interessant mit Splunk eine neue Applikation kennen zu lernen und einen neuen Kurs diesbezüglich grob zu gestalten.

Die persönlichen, wöchentlichen Meetings halfen uns, das Ziel der Arbeit immer genauer zu sehen und den neu entdeckten Möglichkeiten anzupassen. Anstehende Herausforderungen wurden schnell diskutiert und gemeinsam eine Lösung gefunden.

Schade war, dass der fitPC nicht leistungsfähig genug ist und wir somit kein fixfertiges Gerät für den Kurs haben. Doch mit dem fertigen und automatisch startenden Image haben wir die effizienteste Alternativlösung erarbeitet.

Es war toll in einen Teil der Technologie verschiedener Videoportale einzutauchen und zu analysieren, wie sie auf Netzwerkbegebenheiten reagieren. Der Video Stream Analyser funktioniert soweit gut und macht die Auswertung mit Splunk zu einem spannenden Unternehmen. Ich bin gespannt was für Analysemöglichkeiten unser Setup im Kurs künftig eröffnen wird.



## 8.4 Archivdatei Inhalt

---

In der mitgelieferten Archivdatei befinden sich folgende Dateien:

- 01 - Bachelorarbeit.pdf
- 02 - Abstract.pdf
- 03 - Abstract.txt
- 04 - Aufgabenstellung.pdf
- 05 - Poster.pdf
- 06 - Eigenständigkeitserklärung Dürrenberger Lukas.pdf
- 07 - Eigenständigkeitserklärung Ryser Steven.pdf
- 08 - Einverständniserklärung Publikation auf eprints.pdf
- 09 - Urheber- und Nutzungsrechte.pdf
- 10 - Video Stream Analyser Source Code.zip
- 11 - Installationsanleitung.docx
- 12 - Sitzungsprotokolle.zip
- 13 - Persönlicher Bericht Dürrenberger Lukas.pdf
- 14 - Persönlicher Bericht Ryser Steven.pdf
- 15 - Analyse von Videoportalen.xlsx
- 16 - Unterschiede bei Videoportalen.xlsx
- 17 - Gesammelte Beispiel Daten für Splunk.zip
- 18 - Splunk Beispiel Dashboards.zip