

IT Grundlagen

Sven Koesling

Herbst 2015

Inhaltsverzeichnis

Einführung	4
Von Nullen und Einsen	4
Die Entwicklung des Internets	8
Der Vorläufer des Internets: Das Arpanet	8
E-Mail	9
www — die Schweiz beeinflusst die Geschichte des Internets	9
Internet der Dinge	10
Server	11
Hardware	11
Virtualisierung	12
Software	15
Exkurs: Zusammenarbeit und Kommunikation von IT und Fachabteilungen	16
Exkurs: What You Want Is What You Get	16
Datenbanksysteme	17
relationale Datenbanken	17
dokumentorientierte Datenbanken	19
Abstraktion	20
Indexer	20
Dokumentformen	23
Warum nicht Word?	23
html	25
pdf	28
Exkurs: stateless	30
Exkurs: Themen für die Postersession	32
Skriptsprachen	33
serverseitige Skriptsprachen	34
clientseitige Skriptsprachen	35
regular Expressions	35
in medias res	36
Exkurs: Zusammenarbeit und Kommunikation von IT und Fachabteilungen	37
Exkurs: Was passiert bei der Umstellung von KIDS auf RDA?	38

Horizon Report 2015 Library Edition	40
Die Cloud	42
Sicherheit	44

Einführung

04.12.2015: Von Nullen und Einsen, Die Entwicklung des Internets, Server

11.12.2015: Datenbanksysteme

18.12.2015: Dokumentformen, Skriptsprachen, in medias res

08.01.2016: Der Horizon Report 2015, die Cloud, Sicherheit

Von Nullen und Einsen

Es gibt 10 Sorten von Menschen: Diejenigen, die das Binärsystem verstehen,
und die übrigen. (Autor unbekannt)

Computer basieren darauf, dass man in Schaltkreisen den Strom an- bzw. abschalten kann. Es gibt nur die zwei Zustände “AN” und “AUS”. Mathematisch ist das kein Problem, mit jeder Anzahl von Ziffern > 2 lässt sich zählen.

Im täglichen Leben benutzen wir das Dezimalsystem. Wir nehmen die Ziffern von 0 bis 9, und wenn sie aufgebraucht sind, erhöhen wir vorne um eine Stelle.

Im Binärsystem haben zwei Zustände, “AN” und “AUS”, also stehen uns zwei Ziffern zur Verfügung. Zwei Ziffern, 0 und 1 reichen zum Zählen.

Die Folge der ersten neun Binärzahlen...

0, 1, 10, 11, 100, 101, 110, 111, 1000 ...

...und die “Übersetzung”

0	0
1	1
10	2
11	3
100	4
101	5
110	6
111	7
1000	8

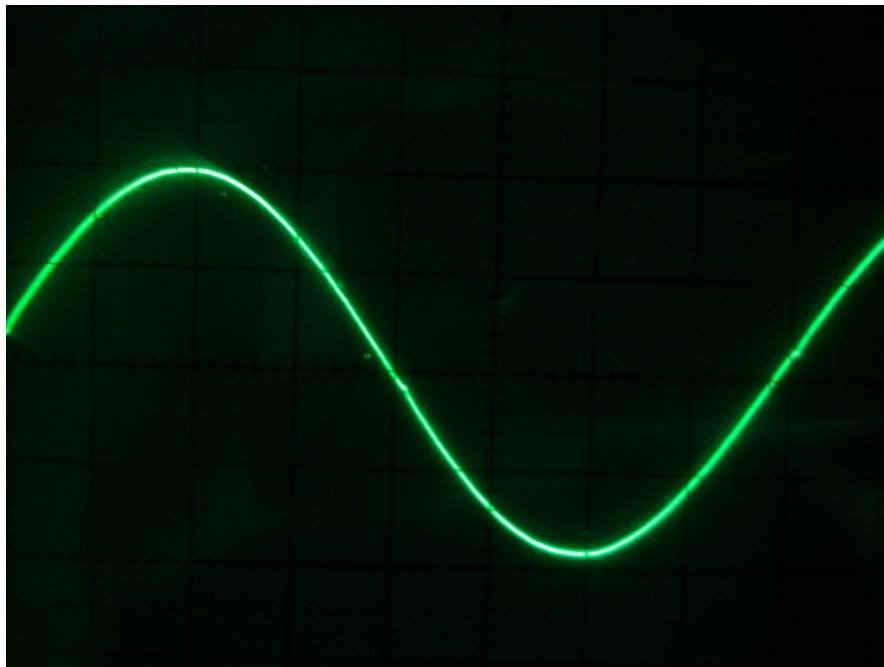


Abbildung 1: Sinuskurve, Quelle: Computer:club², Urheber: Rolf Degen

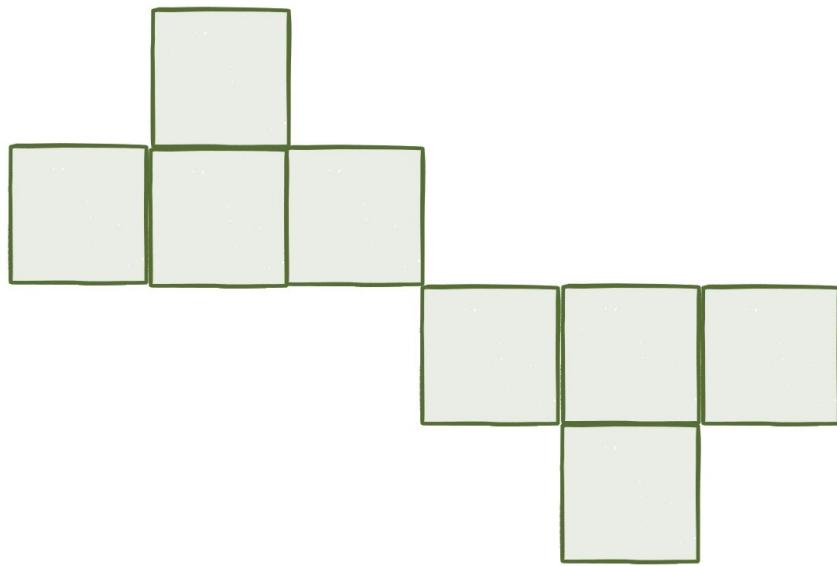
Leibniz schreibt Ende des 17. Jahrhunderts dazu:

...deshalb ist der letzte Tag der vollkommenste und der Sabbat, denn an ihm ist alles geschaffen und erfüllt, und deshalb schreibt sich die 7 111, also ohne Null. Und nur wenn man die Zahlen bloß mit 0 und 1 schreibt, erkennt man die Vollkommenheit des siebenten Tages... (Gottfried Wilhelm Leibniz)

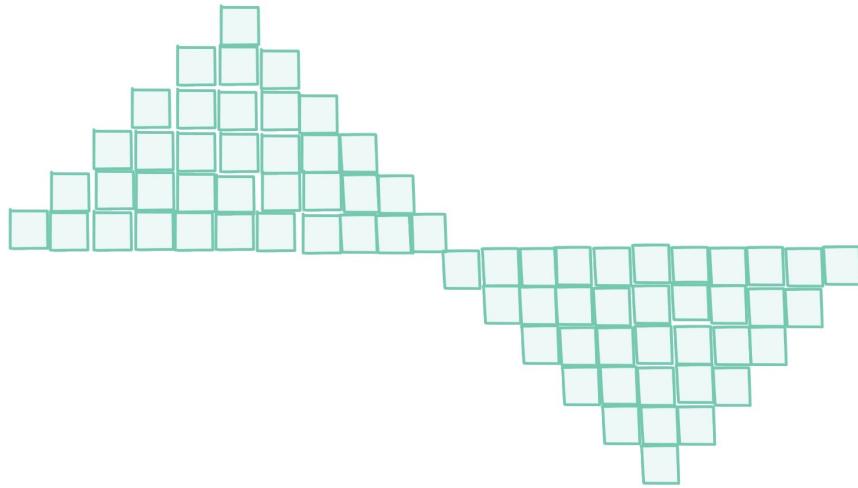
Wie geht das Ganze beispielsweise mit Musik? Man kann sich mit “AN” / “AUS” — Zuständen einer Kurve annähern. Übrigens: Egal, wie leistungsfähig Computersysteme sind, oder noch sein werden, es wird immer eine Annäherung sein. Glücklicherweise ist unser Ohr als Messinstrument so unsensibel, dass Ende der siebziger Jahre die Firma Philipps der digitalen Musikproduktion und –wiedergabe kräftigen Schwung geben konnte: Mit der Einführung des CD-Spielers.

Ein Ton ist eine Schwingung, Sinustöne sind sehr reine Töne, deren Schwingung sehr sauber der Sinuskurve entspricht.

Wir versuchen nun, mit Karos und unseren zwei Zuständen eine solche Kurve zu erzeugen. Nehmen wir an, dass ein ausgefülltes Karo dem Zustand “AN” entspricht, ein leeres Karo dem Zustand “AUS”. Wenn man jetzt ausgefüllte Karos zu einer Pyramide zusammenstellt und dahinter an der Basis genau so eine Pyramide nach unten zeigen lässt, erhält man eine Annäherung an eine Sinuskurve.



Je mehr Informationen wir verwenden, desto feiner wird die Annäherung an die Kurve.



Da unser Auge wie unser Ohr ein begrenzt feines Messinstrument ist, brauchen wir das Karo-Muster nur klein genug zu machen (oder weit genug weg zu nehmen) und können die Treppenstufen der Karos nicht mehr sehen bzw. hören.

Mit "AN" / "AUS" Informationen lässt sich also die Wirklichkeit näherungsweise beschreiben.

Jeh mehr "an" / "aus" Informationen wir einsetzen, desto näher ist das Ergebnis an der Wirklichkeit. Entsprechend steigen aber auch die benötigte Rechenleistung und der Speicherplatzbedarf an.



Ein Bild mit wenig Informationen

Ein Bild mit vielen Informationen

Deshalb rechnet man mit speziellen Algorithmen “nicht benötigte” Informationen aus den Daten heraus.

Auch wenn der Leistungszuwachs von Computern aussergewöhnlich ist, stehen Speicher und Rechenpower nicht unbegrenzt zur Verfügung und sollten ganz normal als Ressource wahrgenommen werden, die man nicht verschwendet.

Übrigens: Zeitungspapier hält 10 – 50 Jahre, ein USB-Stick nur 3 – 10 Jahre. Für digitale Daten sind ausgeklügelte Backup—Systeme dringend notwendig!



Die Entwicklung des Internets

- ARPA
- E-Mail
- www — ein neuer Treiber
- Web Apps, Cloud Services und intelligente Kühlsschränke

Der Vorläufer des Internets: Das Arpanet

Das Internet ist 1969 aus einer Zusammenarbeit des US-Verteidigungsministeriums und verschiedenen Forschungseinrichtungen entstanden. Obwohl sich das Gerücht hält, dass im kalten Krieg eine Technik aufgebaut werden sollte, die im Falle eines Atomschlages die Kommunikationsinfrastruktur erhalten kann, dürfte ein ausschlaggebender Grund für die Entwicklung die bessere Ausnutzung teurer Rechenkapazitäten gewesen sein. Entstanden ist als Vorläufer des heutigen Internets das Arpanet.

So oder so — die Idee ist genial: Man teilt Kommunikation in kleine Päckchen auf und entwickelt Protokolle, die es ermöglichen, dass sich diese Päckchen ihren Weg selbstständig vom Sender zum Empfänger suchen. Dabei gibt es nicht nur eine Leitung von A nach B, sondern ein fein verzweigtes Netzwerk mit unzähligen Knoten. Wenn nun eine Leitung blockiert ist, nimmt das Päckchen einfach einen anderen Weg.

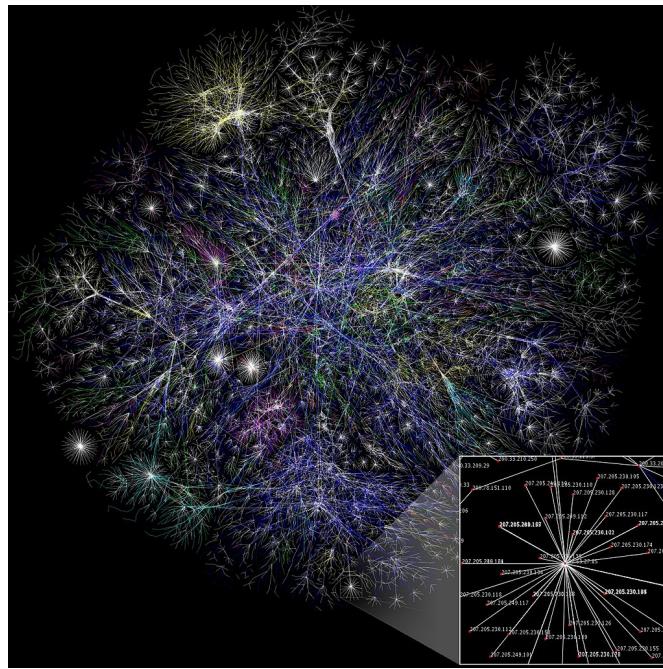


Abbildung 2: Das Internet heute; Quelle: Wikipedia, Urheber: The Opte Project

Übertragen wurden damals übrigens noch keine Webseiten mit Bildern.

E-Mail

Beim Aufbau des Arpanet ging es um Kommunikation von Maschinen. Die Initiatoren konnten sich nicht vorstellen, dass der Austausch von Botschaften irgendeine Rolle in einem Netzwerk von wissenschaftlichen Computern spielen könnte. Aber schon Anfang der siebziger Jahre gab es Techniken, Nachrichten über das Netz auszutauschen, indem man dem Benutzernamen des Adressaten ein @ und den Namen des Computers anfügte.

www — die Schweiz beeinflusst die Geschichte des Internets

Das CERN spielt eine wichtige Rolle bei der Entwicklung des Internets, wie wir es heute kennen. Die Laboratorien des CERN liegen teilweise auf schweizer Gebiet, teilweise auf französischem. Natürlich setzt jedes Land seine eigenen Systeme ein, was es damals unmöglich machte, Texte online auszutauschen.

Mitte der achtziger Jahre nahm sich ein britischer Physiker und Informatiker namens Tim Berners-Lee dieses Problems an und entwickelt mit seinem Kollegen Robert Cailliau ein Konzept für ein weltweites Hypertext-Projekt, das sie 1990 veröffentlichen. Das

daraus entstehende Protokoll “http” und die Auszeichnungssprache “html” sind auch heute noch die Grundlagen des **WorldWideWeb**. Dabei geht es darum Texte über das Netzwerk zur Verfügung stellen zu können.

Die Problematik, Texte universell für verschiedenste System darstellbar zu übertragen, wird auch heute noch deutlich, wenn man z.B. eine Webseite auf einem Smartphone öffnet, die für den Desktop optimiert ist.

Inzwischen sind die Texte um Bilder “bereichert”, Filme werden über das Internet gestreamt und Weltkarten in 3D betrachtet.

Internet der Dinge

Da man nun immer mehr Leistung in immer kleinere Chips packen kann, können kleinste Dinge Funktionen bekommen, für die man früher ganze Rechenzentren benötigte.

Wecker zeigen das Wetter an, Kalender berechnen die Wegzeit automatisch in Alarne mit ein, Navigationssysteme verwenden aktuelle und zu erwartende Verkehrsdaten, um die optimale Route zu bestimmen. Für all das benötigen die Dinge eine Verbindung zum Internet.

Das wirft verschiedene Probleme auf: Sicherheit und Datenschutz sind ein Thema. Aber auch technisch braucht es neue Ansätze. So muss jeder, der gerichtete Informationen bekommen will, eindeutig identifizierbar sein. Die ursprünglich für diesen Zweck gemachte Adressierung hat für das Internet der Dinge viel zu wenig Adressen. Eine neue Technik zur Adressierung ist vorhanden, setzt sich aber nur langsam durch.



Abbildung 3: MacMini Server; Quelle: Gizmodo India

Server

- Hardware
- Virtualisierung
- Software

Hardware

Was ist ein Server? Im Grunde genommen ist jedes Gerät, das Dienstleistungen zur Verfügung stellt, ein Server. Wenn ich von meinem lokalen PC eine Webseite bereit stelle, die jemand anderes von seinem PC aufruft, ist mein PC in dem Moment ein Server.

Ursprünglich war Rechenleistung so teuer, dass es nur einen riesigen Rechner in einem gut gekühlten Keller gab. Terminals ohne eigene Rechenkapazität stellten für eine Anzahl von Nutzern eine Verbindung zu diesem Rechner her, dessen Leistung unter allen Nutzern aufgeteilt wurde. Irgendwann wurde die Technik so billig, dass mit dem Aufkommen der Personal Computers jeder seinen eigenen Rechner auf seinem Schreibtisch hatte.

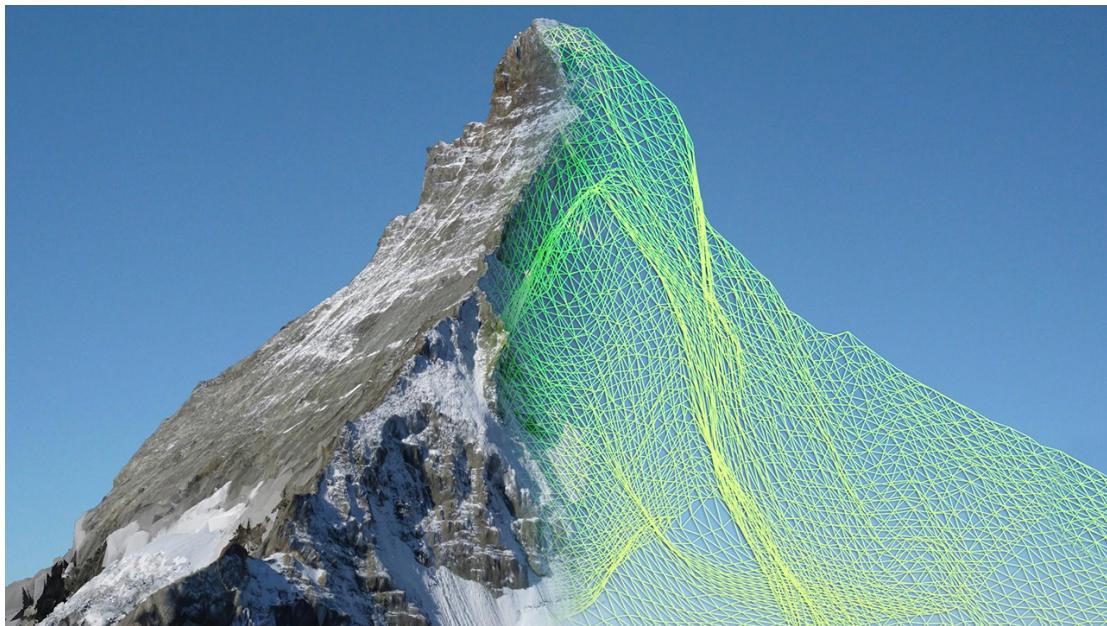


Abbildung 4: Photomontage: Jamani Caillet / © EPFL
<http://actu.epfl.ch/news/the-matterhorn-like-you-ve-never-seen-it/>

Das Moorsche Gesetz besagt dass sich die Rechenleistung alle (je nach Quelle) ein bis zwei Jahre verdoppelt.

Inzwischen haben wir mehr Leistung in einem Mobiltelefon, als noch vor zehn Jahren auf dem Desktop.

Im Jahr 2000 wurden für den Aufbau einer Datenbank für kurze Texte noch 40.000 DM für einen Server der Firma SUN ausgegeben. Der Vorschlag, stattdessen einen Linux-PC für 5.000 DM einzusetzen war revolutionär... und erfolgreich. Tatsächlich waren die PCs so leistungsfähig geworden, dass es keinen Grund mehr gab, das Achtfache zu investieren.

Sogar MacMinis wurden schon benutzt, um den kompletten Webauftritt von Firmen zu realisieren.

Virtualisierung

Hardware ist so billig — warum sollte man einen PC virtualisieren wollen?

Seit Microsoft mit seinen Produkten den PC-Markt beherrscht, stellt sich für Nicht-Windows-Nutzer das Problem, wie man Word-Dokumente öffnen soll. Man glaubt es kaum, aber es gab und gibt immer noch Menschen, die kein Word haben; oder es schlicht nicht benutzen wollen. Aber auch diese Leute bekommen regelmässig Anhänge mit der



berühmten Endung “.docx”. Auch gibt es viel Software, die ausschliesslich für Windows geschrieben wurde.

Schliesslich hatte ein PC genug Leistung für zwei, und so sind kluge Köpfe auf die Idee gekommen, einen PC innerhalb eines PCs zu simulieren. Was bedeutet das? Um diese Frage zu klären, müssen wir uns erstmal bewusst werden, was ein PC ist.

die Komponenten eines PCs

Das Herzstück des PCs ist der Prozessor — der eigentliche Rechner.

Damit man mit ihm kommunizieren kann, gibt es Ein- und Ausgabesysteme. Bis in die achtziger Jahre wurden dafür z.B. Lochkarten gebraucht, heute benutzen wir im Wesentlichen Tastatur und Maus, immer öfter auch den Touchscreen. Monitore und Drucker sind Beispiele für Ausgabesysteme.

Die Daten müssen irgendwo gespeichert werden. Auch hierfür taugt die Lochkarte bzw. ganze Lochbänder, inzwischen abgelöst durch magnetische Medien, optische Datenträger und nichtflüchtige elektronische Speicher.

“Nichtflüchtig” leitet zu einem weiteren Speicher über: Der Prozessor eines PCs ist so schnell, dass er einen besonderen, schnellen Zwischenspeicher benötigt, von dem er

Daten laden und auf dem er seine Ergebnisse ablegen kann. Die Rede ist vom RAM, ein schneller elektronischer Speicher, der allerdings seine Informationen sofort verliert, wenn der Strom abgeschaltet wird.

Auf den meisten Heim-PCs ist heutzutage nicht mehr der Prozessor (CPU) der schnellste im Team. Den Titel hat er an den Grafikprozessor (GPU) abgegeben. Dazu muss man wissen, dass die Spieleentwicklung einer der Treiber bei der technischen Entwicklung von PCs ist. Und für Bilder gilt genauso wie für alles Digitalisierte: Je wirklichkeitsgetreuer die Darstellung sein soll, desto mehr “AN” / “AUS” Informationen brauchen wir, und desto leistungsfähiger muss die Hardware sein.

Verbunden werden all die Komponenten durch das sogenannte Motherboard. Auf ihm sitzt auch ein weiterer kleiner Chip, der beim Einschalten des PCs erstmal die Komponenten sortiert und entscheidet, was gestartet werden soll. Das Problem ist nämlich, dass kein Bauteil vom anderen “weiss”. Wir brauchen eine Software, die die Bauteile miteinander verknüpft, das Betriebssystem. Das Betriebssystem übernimmt vom Chip die Hoheit über den Computer und stellt uns unsere Arbeitsumgebung — im Alltag also den Desktop mit Mail-Programm, Textverarbeitung usw. — zur Verfügung.

Der Clou ist, dass man alle diese Komponenten entweder teilen oder komplett simulieren kann. Wenn ein PC beispielsweise 8GB RAM hat, dann kann man 4GB davon für einen virtuellen PC benutzen. Unserem PC, unserem Betriebssystem stehen dann nur noch 4GB zur Verfügung, die übrigen 4GB “gehören” der virtuellen Maschine.

Man startet also seinen PC, meldet sich an und ruft ein Programm auf, das per Software nun alle Komponenten eines PCs noch einmal simuliert und diesen virtuellen PC startet. Eine Festplatte in diesem “PC” ist dann nur eine grosse Datei auf unserer echten Festplatte. Und in diesem virtuellen PC lässt sich dann ein eigenes Betriebssystem installieren. Auf diesem Weg hat man nun auf einem PC gleichzeitig Windows und Linux zur Verfügung.

Da die Leistungsfähigkeit der Hardware inzwischen so stark gestiegen ist, kann man auf einem Computer gleichzeitig mehrere virtuelle PCs starten.

Vorteile der Virtualisierung

- Da virtuelle Computer nur Dateien auf einer Festplatte sind, kann man sie komplett in einem Backup sichern und quasi auf Knopfdruck wieder herstellen.
- Wenn man für einen Computer kurzfristig mehr Leistung braucht, kann man einem virtuellen Computer einfach per Software mehr RAM oder weitere CPUs zur Verfügung stellen. Das geht teilweise unterbruchsfrei.
- Computer sind selten ausgelastet. Wenn man seinen Bedarf auf virtuelle Maschinen verteilt, ist die Auslastung der echten Systeme besser.

Software

Das Wort Server wird synonym auch für die Software benutzt, die dafür zuständig ist, Services zu erbringen. Wenn wir von einem Webserver sprechen, kann sowohl die (virtuelle) Maschine gemeint sein, die die Webseiten ausliefert, als auch die Software auf der Maschine, die diese Arbeit übernimmt.

Beispiele für Software–Server

- Webserver
- Mailserver
- Dateiserver
- Datenbankserver

Immer übernimmt eine entsprechende Software die Aufgabe, Daten auszuliefern bzw. zu empfangen. Auf einem Hardwareserver können mehrere Softwareserver laufen, auch wenn es im Zuge der Virtualisierung sinnvoll erscheint, für jeden Serverzweck eine eigene virtuelle Maschine zu Verfügung zu stellen. So läuft in der ETH–Bibliothek z.B. das Bibliothekssystem auf einem Server, die Datenbank auf einem anderen.

Exkurs: Zusammenarbeit und Kommunikation von IT und Fachabteilungen

Exkurs: What You Want Is What You Get

Herr K. berichtet der KFZ-Werkstatt erregt über sein Ärgernis. Er ist täglich auf einer serpentinenreichen Landstraße unterwegs. In einem unübersichtlichen Streckenteil hoppelt jedes Mal dieser verdammte Hase über die Straße. Da K. Tiere liebt, weicht er instinktiv aus, was bei nassem Fahrbahnbelag dazu führt, dass er unweigerlich ins Rutschen gerät und einen Baum touchiert. Für Schäden an Stoßstange und Karosserie hat er bereits ein halbes Vermögen ausgegeben.

Werkstattmeister M. nickt verständnisvoll: "Was kann ich da für Sie tun?"

Herr K. : "Könnten Sie mir nicht eine zusätzliche Knautschzone an Stoßstange und Karosserie montieren?"

(Michael Stal, heise.de: "Der Hase und der Baum")

Manche Probleme zwischen Fachabteilungen und IT entstehen durch den Versuch der Fachabteilungen, den richtigen Lösungsansatz vorzugeben. So wird in obigem Beispiel wird keine Anforderung formuliert, sondern bereits ein Lösungsansatz. Dem Auftraggeber fehlt aber technisches Hintergrundwissen, so dass er nicht auf die optimale Lösung kommt.

Anforderungen mit Lösungsansätzen sind vollkommen OK, sofern der Auftraggeber dafür triftige Gründe und das entsprechende technische Wissen hat.

Gefährlich sind aber Bezüge auf den Lösungsraum, falls sie von Auftraggebern mit wenig technischem Know-how stammen. (nochmal Michael Stal, heise.de)

Ich würde es nicht gleich als "gefährlich" bezeichnen, aber wenig zielführend ist es auf jeden Fall.

Die Kunst besteht darin, der jeweils anderen Seite gewisse Kompetenzen einzuräumen.

Das gilt sowohl für Fachabteilungen als auch für IT-Abteilungen! (SK)

Tatsächlich scheinen Informatiker wie auch Informationswissenschaftler einen eher speziellen Blick auf selbstverständliche Dinge zu haben — zumindest sieht es für den jeweils anderen so aus, oder?

Datenbanksysteme

Warum Datenbanken?

- Speicherplatz
- gleichzeitiger Zugriff durch viele Nutzer

Da Speicher früher sehr teuer war, entwickelte man Strategien, um Speicherplatz zu sparen. Ein gutes Beispiel sind Normalisierungen bei relationalen Datenbanken.

relationale Datenbanken

Nehmen wir zur Verdeutlichung eine Versicherung, die die Adressen ihrer Kunden in einer Datenbank speichern möchte. Es wird eine grosse Tabelle angelegt, je eine Zeile pro Kunde. Dabei stellt sich heraus, dass die Versicherung 1000 Kunden hat, die in der Hauptstrasse wohnen. Das bedeutet, dass wir 1000 Mal den Speicherplatz für das Wort “Hauptstrasse” benötigen.

Eine Kundentabelle

	id	Nachname	Vorname	Strasse	Stadt
	1	Muster	Hans	Hauptstrasse	Zürich
	2	Meier	Heinrich	Hauptstrasse	Zürich
	3	Müller	Hubert	Hauptstrasse	Zürich
	4	Schulze	Herbert	Hauptstrasse	Zürich

Wenn man nun die Strassen in eine extra Tabelle schreibt, eine Zeile pro Strasse, dann muss man in der Tabelle der Kunden nur die ID der Strasse hinterlegen. Wenn die Hauptstrasse z.B. die ID 1 hat, dann benötigen wir nur noch 1000 Mal den Speicherplatz für den Integer 1 — wesentlich weniger, als für den String “Hauptstrasse”. Analog kann man z.B. auch mit Städten verfahren.

Tabelle der Strassen

	id	Strasse
	1	Hauptstrasse
	2	Nebenstrasse
	3	Seitenstrasse

Tabelle der Städte

	id	Stadt
	1	Zürich
	2	Basel
	3	Bern

Tabelle mit Relationen

	id	Nachname	Vorname	Strassen_ID	Stadt_ID
	1	Muster	Hans	1	1
	2	Meier	Heinrich	1	1
	3	Müller	Hubert	1	1
	4	Schulze	Herbert	1	1

Speicherplatzbedarf

```
-rw-rw-r-- 1 sk sk 37K Dez 9 16:32 testdaten_full.csv
-rw-rw-r-- 1 sk sk 20K Dez 9 17:02 testdaten_normalisiert.csv
```

Eine relationale Datenbank dahingehend zu optimieren, dass es möglichst wenig Redundanzen gibt, nennt man “normalisieren”. Es gibt fünf Normalformen, und für die Datenbank-Spezialisten war es eine komplexe Aufgabe, Daten optimal zu normalisieren.

Die Normalisierung hat auch Nachteile. So liegen nicht alle Informationen in einer Tabelle, was Abfragen komplexer macht. Damit einher geht ein höherer Leistungsbedarf des Servers.

Felder nicht atomar

```
select substr(z103_rec_key,6,9) || 'EHO60'
from eho60.z103
where substr(z103_lkr_text_n,1,3) = 'E04'
and z103_lkr_library = 'EBI01'
and substr(z103_rec_key,1,5) = 'EHO60'
;
```

Ein Nachteil von relationalen Datenbanken ist die feste Größe und Anzahl von Feldern. Wenn ich beispielsweise für die Tabelle der Kunden die Spalten

Vorname Nachname Strasse Hausnummer Postleitzahl Ort

festgelegt habe, und die Versicherung sich entscheidet, ein internationales Geschäft aufzubauen, ist es einiger Aufwand, die Spalte “Land” hinzufügen.

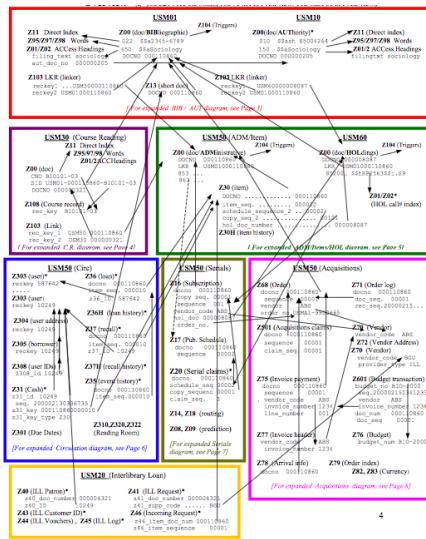


Abbildung 5: Relationen in der DB von Aleph

feste Grösse

Das Feld für Inventarnummern darf in Aleph nicht mehr als 20 Zeichen haben.

und wieder: Kodierung von Text

UTF8-Codierungszeichen zaehlen einzeln, auch wenn daraus ein einziges Unicode-Zeichen entsteht.

Ein Beispiel dafuer ist das “Ä”, das im Inventarnummernfeld zwei VARCHAR2 Zeichen aufbraucht, weil es aus 0xc3 und 0x84 besteht.

Ich habe nicht geschaut, ob es noch weitere solche Fälle gibt. [Mathias Weyland]

dokumentorientierte Datenbanken

```
{
  "_id": 1,
  "Vorname": "Sven",
  "Nachname": "Koesling",
  "email": "sven.koesling@library.ethz.ch"
}
```

ein weiteres Dokument in derselben collection

```
{ "_id": 2,  
  "Vorname": "Harry",  
  "Nachname": "Hirsch",  
  "email": "hh@lustich.com"  
  "Beruf": "Reporter"  
}
```

noch ein Dokument in derselben collection

```
{ "_id": 3,  
  "Vorname": "Martha",  
  "Nachname": "Graham",  
  "Beruf": ["Tänzerin", "Choreographin"]  
}
```

und noch ein weiteres Dokument in derselben collection

```
{ "_id": 4,  
  "Vorname": "Niemand",  
  "Nachname": "Nixda",  
  "email": "nn@none.org"  
  "Hund": {  
    "Name": "Nero",  
    "Geschlecht": "male"  
  }  
}
```

Abstraktion

Indexer

Abbildung 6: Suche nach “Landquart” in einem aktuellen Bibliothekskatalog (420.000 Dokumente): 1072 Treffer – ohne Facetten: completed in 2845ms

The screenshot shows a library catalog interface with the following details:

- Header:** Includes a logo with a blue and yellow shield, navigation links like "Ihr Bibliothekskonto", "Neustart", "Hilfe", and "Suche".
- Search Bar:** Shows the search term "Landquart" and search parameters: "Alle auswählen", "Auswahl entfernen", "Suche modifizieren", and "Filtern".
- Section Header:** "Ergebnisse" (Results).
- Text:** "für "Stichwort= Landquart"; Sortiert nach: Jahr, dann Autor
Sätze 1 - 10 von 1072"
- Table:** A grid of 10 search results. The columns are labeled "#", "Autor", "Titel", "Jahr", "Bestand / davon ausgeliehen", and "Photo".

#	Autor	Titel	Jahr	Bestand / davon ausgeliehen	Photo
1	Gantenbein, Köbi	Das Tor zu Graubünden	2010		
2	Radio- und Fernsehgesellschaft der Deutschen und der Rätoromanischen Schweiz (Zürich)	[am Bündner Ländlerkapellen-Treffen in Landquart] [Ton]	2009	CHUR-Kantonsbibliothek(2 / 0)	
3	Schmid, Christian	Die Bahn fährt durch den Unterschnitt	2009		
4	Uffer, Rita	Luzia Vonmoos-Simonett, Igis [Ton]	2009	CHUR-Kantonsbibliothek(1 / 0)	
5	Vischer, Daniel	Der Bündner Theologe Lucius Pol und die Landquarkorrektion	2009	CHUR-Kantonsbibliothek(1 / 0) CHUR-Staatsarchiv(1 / 0)	
6	ÖKK (Landquart)	ÖKK dossier per aziende	2009		
7	ÖKK (Landquart)	ÖKK Dossier für Unternehmen	2009	CHUR-Kantonsbibliothek	
8		Alles, was in der Ostschweiz gehegt und gepflegt wird. [Bildmaterial]	2009		
9	Brunold-Bigler, Ursula	Arbeiterschaft und Kapuziner	2008	CHUR-Kantonsbibliothek(2 / 0) CHUR-Staatsarchiv(1 / 0) CHUR-Theologische Hochschule(1 / 0)	
10	Herrmann, Arno	Eine Kirche, die redet ...	2008	CHUR-Staatsarchiv(1 / 0)	

Abbildung 7: Die gleiche Suche mit einem Indexer: 1.700 Treffer mit drei Facetten: completed in 792ms

Bibliographisches Portal Graubünden
eine Seite Ihrer Kantonsbibliothek



Regionaler Einstieg Bündner Personen Autorenregister Epochen Bibliographie 2009 Feedback

→ Albula Bernina Hinterhein Imboden Inn Landquart Maloja Moesa Plessur Prättigau / Davos Surseva

Ergebnis für "Landquart"
(1700 Treffer)

« Previous 1 2 3 4 5 6 7 8 9 ... 113 114 Next »

Kreis

Fünf Dörfer
Maienfeld

Autoren (47)

Ferdmann, Jules (2)
Hartmann, Reto (4)
Härtli, Peider (2)

Urheber (31)

Elmer-Cantieni, Anna Maria (2)
Mooser, Ueli (3)
Tschümperlin, Beat (8)
Weber, Johann (2)

Schlagworte (26)

Eisenbahnlinie : Landquart-Davos (7)
Eisenbahnlinie : Landquart-Davos-Filisur (2)
Graubünden (3)
Iglis (3)
Landquart (29)
Landquart (Fluss) (2)

"...und wenn nötig, dieselben zu allem Fleiss anhalten"

Autor / Urheber: Bartli, Hans
Vorkommen: Dorfchronik Jenins..., 11(1998); S. 26-30, 11/1998/26-30, 000150885
ISBN: 0
geograph. Schlagwort: Jenins
zum [Katalog](#)

"Alligator Malans" vor dem Unihockey-Eurocup - Start

Vorkommen: Bündner Zeitung, Chur, 1997/12/30, p. 19, 1997/12/30/19, 000048258
ISBN: 0
zum [Katalog](#)

"Ausgelöffelt" : "Hirschen" in Maienfeld

Vorkommen: Südoestschweiz, Chur, 2002/05/17, Apropos, 2002/05/17, 000167664
ISBN: 0
zum [Katalog](#)

"Bock auf Rock!"

ISBN: 0
geograph. Schlagwort: Zizers
zum [Katalog](#)

"Cucci Travel" Reisebüro neu in Landquart

Vorkommen: Bündner Zeitung, Chur, 1997/3/1, p. 4, 1997/3/1/4, 000048258
ISBN: 0
zum [Katalog](#)

"Das Repertoire möglichst breithalten"

Dokumentformen

Dokumentformen

- Warum nicht Word?
- html
- pdf

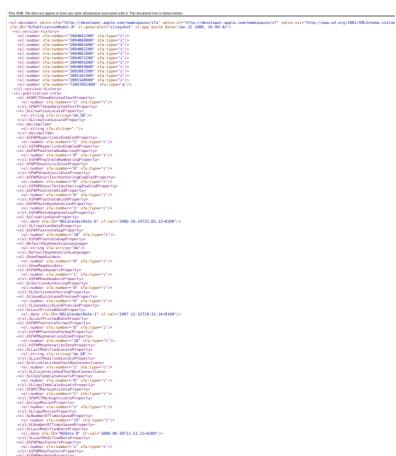
Warum nicht Word?

Das Web ist eine unvorstellbar große Sammlung von Dokumenten. Wenn man also im Internet surft, guckt man sich Dokumente an, die auf anderen Rechnern liegen. Nun haben doch alle Word – warum also ein weiteres Format (html)? Es gibt dazu viele Argumente:

1. Dokumentgrößen

 dokument.doc	28 KB
 dokument.docx	12 KB
 dokument.html	4 KB

die Dokumentauszeichnung am Beispiel “Pages”



(noch 12 mal soviel...)

der Dokumentinhalt am Beispiel “Pages”

The screenshot shows the raw XML source code of a Microsoft Word document. The code is highly nested and structured, reflecting the complex nature of Word's document format. It includes elements for text, tables, images, and various styling and layout properties. The code is presented in a monospaced font, with different colors used to highlight different XML tags and attributes.

Bei der Übertragung über das Netz ist die Datei–Größe trotz “dicker Leitungen” nach wie vor entscheidend: Ein Word- Dokument mit dem gleichen Inhalt ist sieben mal (Word97-2003) bzw. dreimal (Word2007) grösser, als eine html-Datei gleichen Inhalts. html ist ein Format, das sehr schlank ist und schnell übertragen werden kann.

2. frei verfügbares Format

Ein proprietäres Format wie .doc wird laufend verändert, wobei diese Veränderungen nicht dokumentiert werden. Wer das Dokument lesen will, muss die Software zum Lesen kaufen (können...).



Webseiten sollten aber von jedem weltweit betrachtet werden können.

Ein weiterer Unterschied: Dokumente von Textverarbeitungen sind dafür gemacht, dass man sie leicht editieren kann. html- Dokumente dagegen sollen nach ihrer Erzeugung zunächst mal nur angezeigt, aber nicht bearbeitet werden.

Der Webbrowser schickt dazu eine Anfrage an den Server “Reich mir mal die Seite /dokument.html rüber!”. Wenn die Seite vorhanden ist, gibt der Server sie heraus, sonst wird eine Fehlermeldung angezeigt. Der Webbrowsers rückt die Seite raus, der Browser

zeigt die Seite an - sobald man aber weiter surft, ist sie schon wieder vergessen (grund-sätzlich... Browser speichern heute Seiten in einem sogenannten Cache, damit sie beim nächsten Aufruf schneller kommen.). Bearbeiten kann man die Seite nicht. Mit speziellen Tools lässt sich der empfangene Code natürlich editieren, dann aber nur auf dem eigenen Rechner betrachten. Der Server wird es nicht erlauben, dass jeder einfach seine Änderungen auf ihm abspeichert.

Und schliesslich gibt es für Textverarbeitungen fest definierte Dokumentgrößen (A4 z.B.), Webseiten müssen aber auf den unterschiedlichsten Monitorgrößen angezeigt werden. Das bedeutet, dass z.B. Zeilenumbrüche flexibel sein müssen.

html

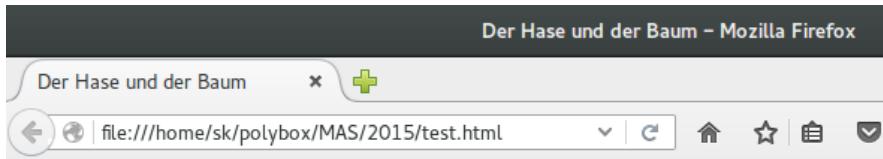
html ist zunächst einmal keine Programmiersprache. Man kann keinen Rechner damit füttern und ihm das Ergebnis von $2 + 2$ entlocken.

html ist eine Auszeichnungssprache, die dazu dient, Inhalt zu formatieren. Dazu versieht der Programmierer den strukturierten Inhalt mit Tags — eines zum Anfang und eines zum Ende des jeweiligen Bereichs. So kennzeichnen beispielsweise `<p> </p>` einen Absatz. Zwischen den Tags steht dann der Inhalt.

eine einfache Seite

```
<html>
  <head>
    <title>Der Hase und der Baum</title>
  </head>
  <body>
    <h1>Der Hase und der Baum</h1>
    <h2>Kapitel 1: Der Hase</h2>
    <p>Meister Lampe hoppelt über ein Feld.</p>
    <h2>Kapitel2: In der Werkstatt</h2>
    <p>Herr K. bestellt eine Knautschzone.</p>
  </body>
</html>
```

...sieht so aus:



Der Hase und der Baum

Kapitel 1: Der Hase

Meister Lampe hoppelt über ein Feld.

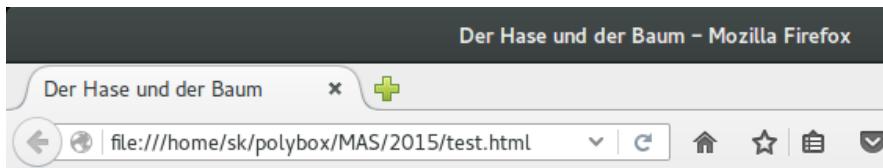
Kapitel2: In der Werkstatt

Herr K. bestellt eine Knautschzone.

...mit Umlauten:

```
<html>
  <head>
    <meta charset="utf-8" />
    <title>Der Hase und der Baum</title>
  </head>
  <body>
    <h1>Der Hase und der Baum</h1>
    <h2>Kapitel 1: Der Hase</h2>
    <p>Meister Lampe hoppelt über ein Feld.</p>
    <h2>Kapitel2: In der Werkstatt</h2>
    <p>Herr K. bestellt eine Knautschzone.</p>
  </body>
</html>
```

...sieht so aus:



Der Hase und der Baum

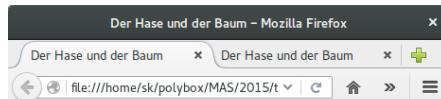
Kapitel 1: Der Hase

Meister Lampe hoppelt über ein Feld.

Kapitel2: In der Werkstatt

Herr K. bestellt eine Knautschzone.

So soll es sein:



Der Hase und der Baum

Kapitel 1: Der Hase

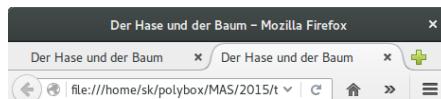
Meister Lampe hoppelt über ein Feld.

Kapitel2: In der Werkstatt

Herr K. bestellt eine Knautschzone.

 Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

So soll es nicht sein!



Der Hase und der Baum

Kapitel 1: Der Hase

Meister Lampe hoppelt über ein Feld.

Kapitel2: In der Werkstatt

Herr K. bestellt eine Knautschzone.

 Lorem ipsum dolor sit amet, consectetur adipisicing elit, labore et dolore magna aliqua. Ut enim ad minim veniam laboris nisi ut aliquip ex ea commodo consequat. Duis au voluptate velit esse cillum dolore eu fugiat nulla pariatur, proident, sunt in culpa qui officia deserunt mollit anim id

pdf

Wenn man ein Dokument mit festem Layout veröffentlichen möchte, bietet sich PDF an. Es ist frei verfügbar, kann also von jedem gelesen werden, und sieht immer gleich aus. Wie auch Webseiten lässt es sich (eigentlich) nicht vom Empfänger verändern.

Wieder die Dokumentgrösse:**Kursus — Übersicht**

04.12.2015: Von Nullen und Einsen, Die Entwicklung des Internets, Server

11.12.2015: Datenbanksysteme

18.12.2015: Dokumentformen, Skriptsprachen, in medias res

08.01.2016: Der Horizon Report 2015, die Cloud, Sicherheit

Sven Koesling (ETH-Bibliothek)

IT Grundlagen

2 / 54

als Bild: 72.2 kB (72'224 Bytes) als PDF: 35.9 kB (35'880 Bytes)

Fazit: Welche Form wofür?

Textverarbeitung: Alles zum Weiterverarbeiten

html: Inhalt geht über Form

PDF: Form soll erhalten bleiben

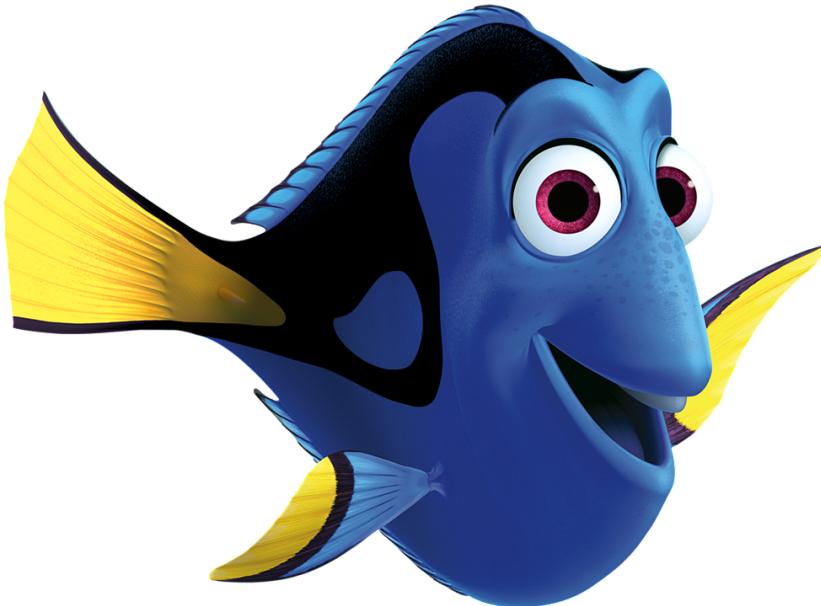
alles Quatsch?

Satz 1. *Im Web liest doch niemand mehr. Da sollte man nur knappe Infos unterbringen.*

Exkurs: stateless

Ein Problem bei der Entwicklung von Web–Apps war, dass ein Webserver eigentlich nichts vom User weiss. Er liefert die geforderte Webseite aus und hat dann den User schon wieder “vergessen”.

http is stateless



HTTP is a stateless protocol. A stateless protocol does not require the server to retain information or status about each user for the duration of multiple requests.

But some web applications may have to track the user’s progress from page to page, for example when a web server is required to customize the content of a web page for a user. Solutions for these cases include:

- the use of HTTP cookies.
- server side sessions,
- hidden variables (when the current page contains a form), and
- URL-rewriting using URI-encoded parameters, e.g., /index.php?session_id=some_unique_session_code.

(wikipedia)

Um das zu ändern gibt es verschiedene Techniken wie z.B. Cookies und Sessions. Dabei handelt die Web-App einen eindeutigen Identifier aus und kann so wiederkehrende User identifizieren. „Dank“ der Werbeindustrie wurden diese Techniken so weit optimiert, dass wir im Internet gläsern sind. Man kann feststellen, woher wir kommen, welche Seiten wir gesehen haben, welche wir wegklicken, wie lange wir wo verweilt haben und, und, und.

privacy?

You have zero privacy anyway. Get over it.
(Scott McNealy, Sun Microsystems, 1999)

Exkurs: Themen für die Postersession

Google und Bibliotheken

- Welche Gemeinsamkeiten und welche Unterschiede bestehen in den Anforderungen an Google und an eine Bibliothek?
- Auf welche Datenbanktechnologien setzt Google?
- Welches Business wäre eventuell für den Wissenstransfer besser geeignet, als Google?

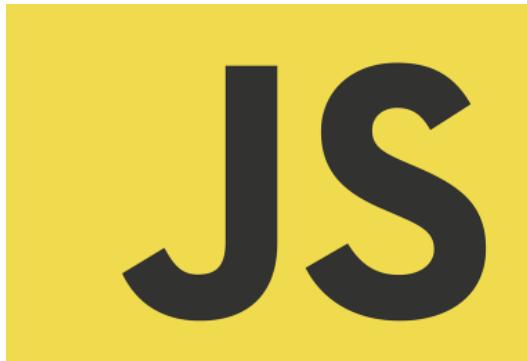
Die Zusammenarbeit von Fachabteilungen und IT

- Beschreiben Sie eine typische Situation mit Differenzen aus Sicht der Fachabteilung.
- Beschreiben Sie die gleiche Situation so, wie Sie sich die Sicht der IT vorstellen.
- Was wäre aus Ihrer Sicht der Idealzustand?

Der Horizonreport 2015 Library Edition

- Beschreiben sie die Auswirkungen einer möglichen zukünftigen technischen Entwicklung auf Bibliotheken.
- Welche Rolle spielt die Strategie in dieser Entwicklung?
- Was müsste man aus Ihrer Sicht für Bibliotheken erfinden?

Skriptsprachen



Während man mit Auszeichnungssprachen wie z.B. html oder L^AT_EX Dokumente formatiert, dienen Skriptsprachen wie Perl, Ruby oder Javascript dazu, einfache, wiederkehrende Aufgaben am Computer zu übernehmen.

Beim Publishing von Aleph erhalten wir beispielsweise unzählige komprimierte Dateien, die ihrerseits unzählige XML-Dateien enthalten. Jede dieser XML-Dateien enthält einen Datensatz.

Das erste Programm

Number of Operation	Statement of Operation	Variables acted upon	Variables receiving results	Indication of change in the value of any Variable	Statement of Results										Working Variables		Result Variables			
					V ₁	V ₂	V ₃	V ₄	V ₅	V ₆	V ₇	V ₈	V ₉	V ₁₀	V ₁₁	V ₁₂	V ₁₃	V ₂₁	V ₂₂	V ₂₃
1	= V ₂ × V ₃	V ₄	V ₄ , V ₅ , V ₆	{V ₂ = V ₃ } = 2n	1	2	n	2n	2n	2n								B ₁	B ₂	B ₃
2	- V ₄ = V ₅	V ₂	V ₄	{V ₄ = V ₅ } = 2n - 1	1															
3	= V ₅ + V ₆	V ₂	V ₅	{V ₅ = V ₆ } = 2n + 1	1															
4	= 2V ₆ + V ₂	V ₁₁	V ₆	{V ₆ = V ₂ } = $\frac{2n+1}{2}$																
5	= V ₁₁ + V ₂	V ₁₁	V ₁₁	{V ₁₁ = V ₂ } = $\frac{2n+1}{2} + \frac{2n+1}{2}$	2															
6	- V ₁₁ = V ₁₁	V ₁₁	V ₁₁	{V ₁₁ = V ₁₁ } = - $\frac{2n+1}{2}$ = A ₀																
7	- V ₁₁ = V ₁₁	V ₁₀	V ₁₀	{V ₁₀ = V ₁₁ } = n - 1 (n > 3)	1															
8	= V ₂ + V ₅	V ₇	V ₇	{V ₂ = V ₅ } = 2 + 0 = 2	2															
9	= V ₆ + V ₇	V ₁₁	V ₁₁	{V ₆ = V ₇ } = 2n - 2 = A ₁																
10	= V ₂ × V ₁₁	V ₁₂	V ₁₂	{V ₂ = V ₁₁ } = 2n - 1 = B ₁ A ₁																
11	= V ₁₂ + V ₁₁	V ₁₂	V ₁₂	{V ₁₂ = V ₁₁ } = - $\frac{2n-1}{2}$ = B ₁ A ₁																
12	- V ₁₀ = V ₁₁	V ₁₀	V ₁₀	{V ₁₀ = V ₁₁ } = n - 2 (n > 2)	1															
13	{= V ₆ - V ₅	V ₆	V ₆	{V ₆ = V ₅ } = 2n - 1	1															
14	{= V ₁ + V ₇	V ₆	V ₆	{V ₁ = V ₇ } = 2 + 1 = 3	1															
15	{= V ₆ + V ₅	V ₈	V ₈	{V ₆ = V ₅ } = $\frac{2n-1}{2}$																
16	= V ₆ × V ₁₁	V ₁₁	V ₁₁	{V ₆ = V ₁₁ } = 2n - 1 = B ₂ A ₁																
17	- V ₆ - V ₅	V ₆	V ₆	{V ₆ = V ₅ } = 2n - 2	1															
18	{= V ₁ + V ₇	V ₇	V ₇	{V ₁ = V ₇ } = 3 + 1 = 4	1															
19	{= V ₆ + V ₅	V ₉	V ₉	{V ₆ = V ₅ } = B ₂ A ₂																
20	= V ₆ × V ₁₁	V ₁₁	V ₁₁	{V ₆ = V ₁₁ } = 2n - 2 = B ₂ A ₂																
21	= V ₂ × V ₁₁	V ₁₂	V ₁₂	{V ₂ = V ₁₁ } = B ₁ A ₂																
22	= V ₁₂ + V ₁₁	V ₁₃	V ₁₃	{V ₁₂ = V ₁₁ } = A ₀ + B ₁ A ₂ = B ₁ A ₃																
23	- V ₁₀ = V ₁₁	V ₁₀	V ₁₀	{V ₁₀ = V ₁₁ } = n - 3 (n > 1)	1															
24	= V ₁₃ + V ₂₂	V ₂₃	V ₂₃	{V ₁₃ = V ₂₂ } = B ₂ A ₃																
25	= V ₁ + V ₃	V ₃	V ₃	{V ₁ = V ₃ } = n + 1 = 4 = 5 by a Variable-card. by a Variable-card.	1															

Here follows a repetition of Operations thirteen to twenty-three

Ein Programm

Das folgende Skript (Auszug) dient zum Durchsuchen von tar-files:

```
Dir.entries("./").each do |tarname|
  if tarname =~ /tar\.gz$/
    tarinhalt = %x(tar -tzf #{tarname})
    tarinhalt.each_line do |dateiname|
      dateiname.chomp!
      if dateiname =~ /#{suchstring}/
        puts "#{$<}" kommt in #{$<} im Archiv #{tarname} vor"

        puts "Soll die Datei extrahiert werden: j/N"
        antwort = gets
      end
    end
  end
end
```

...

serverseitige Skriptsprachen

Die Einteilung in server— und clientseitige Skriptsprachen ist ein bisschen willkürlich, macht aber bestimmte Konzepte gut deutlich. Grundsätzlich könnten alle Skriptsprachen sowohl auf dem Client als auch auf dem Server vorhanden sein.

Hier soll es aber darum gehen, dass — je nachdem, auf welcher Seite das Skript läuft — bestimmte Funktionen möglich sind, andere eben nicht.

clientseitige Skriptsprachen

dienen dazu, mit dem Server zu kommunizieren, Funktionen auf dem Client auszuführen, die Webseite beim User zu manipulieren

regular Expressions

in medias res

Wir setzen das Gelernte zusammen

Wie entstehen dynamische Webseiten?

Exkurs: Zusammenarbeit und Kommunikation von IT und Fachabteilungen

Informatiker sind merkwürdig

Missverständnis

Meine Frau trug mir auf: "Liebling, bitte geh' zum Volg und kaufe eine Flasche Milch — wenn sie Eier haben, kaufe sechs."

Ich kam mit sechs Flaschen Milch zurück.

Sie: "Warum zum Teufel hast Du sechs Flaschen Milch geholt?"

Ich: "Weil sie Eier hatten!?"

RDA-63 STP/Aleph Test: GND (neuen Satz erstellen, Satz ableiten, abspeichern)	wichtig
RDA-63 STP/Aleph Test: Indexierung testen (Wort- und Indexsuche)	wichtig
RDA-63 STP/Aleph Test: Vorhandene Sätze duplizieren und anpassen	wichtig
RDA-63 STP/Aleph Test: Neue Titelaufnahme erstellen	wichtig
RDA-63 STP/Aleph Test: Fremddatenübernahme	wichtig
STP: Test der RDA/GND Konfiguration Haupttask	wichtig
RDA-54 8. Systemabnahme und -freigabe	wichtig
RDA-54 7. Bearbeitung der Rückmeldungen	wichtig
RDA-54 6. Testen der Konfiguration	wichtig
RDA-54 5. Abschließende Arbeiten	wichtig
RDA-54 4. Übernahme der Konfiguration: alephe und ext09/tab	wichtig
RDA-54 3. Übernahme der Konfiguration: ebi01/pc_tab/catalog	wichtig
RDA-54 2. Übernahme der Konfiguration: ebi01/tab	wichtig
RDA-54 1. Vorbereitende Arbeiten vor Konfigurationsübernahme	wichtig
2.1.3 STP - RDA/GND Haupttask	wichtig
RDA-11 OAI-Sets anpassen	wichtig
RDA-9 alephe/tab tab_base Änderungen übernehmen	todo
u-Baum-Verzeichnis ids18	wichtig
RDA-9 EXT09 Tabellen konfigurieren	wichtig
RDA-9 z39_gate Änderungen an Wörldcat übernehmen	wichtig
RDA-9 pc_tab/catalog/html Feldhilfen	wichtig
RDA-10 Gonin-Routinen einrichten und aktualisieren	wichtig
RDA-9 Erweiterungen tab33.lng f. RDA	todo
RDA-21 modifizierter p-manage-102	wichtig
RDA-10 tab_aut für GNDS	wichtig

Exkurs: Was passiert bei der Umstellung von KIDS auf RDA?

RDA

Die Umstellung von KIDS auf RDA

und die Ablösung der EAF10 durch GND

- **NVZ:** Drei FTE seit Oktober allein für Schulungen
- **BIT:** Drei FTE seit Mai für Konzeptentwicklung, Konfigurationen, Tests, Indexierungen
- **diverse Abteilungen:** Teilnahme an Schulungen, Unterstützung bei Tests

Auszug aus der Aufgabenliste

Konfigurationen

- Das Format zum Regelwerk
- Automatisierungen

- Schnittstellen
- Im- und Exportskripte

Horizon Report 2015 Library Edition

Der Horizon Report 2015 Library Edition

Der Horizon Report 2015 Library Edition

Im Horizon Report 2015 Library Edition haben sich Informations- und Bibliotheksspezialisten Gedanken um zukünftige Entwicklungen und deren möglichen Einfluss gemacht (siehe auch: Innovationsblog der ETH-Bibliothek).

Dabei geht es um drei grosse Themenkomplexe:

1. für Forschungsbibliotheken wichtige technologische Entwicklungen
2. Trends, die die Umsetzung beschleunigen
3. und Herausforderungen, die diese hemmen

Important Developments in Technology for Academic and Research Libraries

- Time-to-Adoption Horizon: One Year or Less
 - Makerspaces
 - Online Learning
- Time-to-Adoption Horizon: Two to Three Years
 - Information Visualization
 - Semantic Web and Linked Data
- Time-to-Adoption Horizon: Four to Five Years
 - Location Intelligence
 - Machine Learning

Trends Accelerating Technology Adoption in Academic and Research Libraries

- Long-Term Impact Trends: Driving technology adoption in academic and research libraries for five or more years
 - Increasing Accessibility of Research Content
 - Rethinking Library Spaces
- Mid-Term Impact Trends: Driving technology adoption in academic and research libraries over the next three to five years
 - Evolving Nature of the Scholarly Record
 - Increasing Focus on Research Data Management
- Short-Term Impact Trends: Driving technology adoption in academic and research libraries over the next one to two years
 - Increasing Value of the User Experience
 - Prioritization of Mobile Content and Delivery

Challenges Impeding Technology Adoption in Academic and Research Libraries

- Solvable Challenges: Those that we understand and know how to solve
 - Embedding Academic and Research Libraries in the Curriculum
 - Improving Digital Literacy
- Difficult Challenges: Those that we understand but for which solutions are elusive
 - Competition from Alternative Avenues of Discovery
 - Rethinking the Roles and Skills of Librarians
- Wicked Challenges: Those that are complex to even define, much less address
 - Embracing the Need for Radical Change
 - Managing Knowledge Obsolescence

Die Cloud

Über die Cloud ist soviel Halbwissen verbreitet worden, dass ich empfehle, das Thema nochmal ganz von vorne anzudenken!

Die Cloud

The computer industry is the only industry, that is more fashion-driven, than women's fashion. Maybe I'm an idiot, but I have no idea what anyone is talking about. What is it? It's completely gibberish. It's insane. When is this idiocy going to stop? (Larry Ellison auf der "Oracle Open World 2008")

Grundsätzlich geht es beim Cloud—Computing um über die ganze Welt verteilte Daten und Rechenkapazitäten. Dabei spielt es durch die zusammengeschaltete Menge an Rechnern keine Rolle, wenn einzelne Systeme mehr oder weniger Leistung bringen oder gar komplett ausfallen. Auch die Daten sind so intelligent zerlegt und verteilt, dass der Ausfall eines Datencenters unerheblich ist. Exzellente Beispiele sind Universitätsprojekte wie seti@home oder folding@home.

Meistens wird beim Outsourcing ein externes Rechenzentrum irrtümlich als Cloud bezeichnet. Tatsächlich bedeutet Cloud eigentlich, dass die Daten nicht mehr (ausschliesslich) im eigenen Haus liegen, aber eben nicht nur in einem Rechenzentrum, sondern verteilt auf viele.

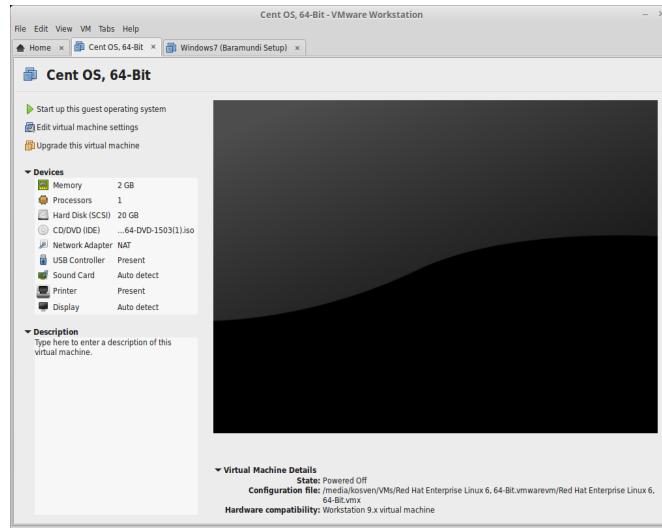
Services in der Cloud

- IaaS
- SaaS
- PaaS

Larry Ellison hat vollkommen Recht: Keiner dieser Services setzt die Cloud voraus! Aber sie alle können davon profitieren.

IaaS

Infrastructure as a Service



SaaS

Software as a Service

Sitzungsnummer		16
Datum	27.08.2015, 14:00 – 15:15 Uhr, HG H 59.2	
Anwesende	Judith Bissegger, Barbara Wittwer, Marianne Wolff	
Entschuldigt	Erika Pfister, Madeleine Völmin	

Ergebnisse

- Nachbearbeitung E-Book Anreicherung: Analyse der angereicherten Daten und notwendige Bereinigungen, sowie Erstellung von SYS-Nr.-Listen für MDM
- Inhalte der Folder 009906907 genauer analysiert und Gruppen von fehlerhaften Codes komplett
- Prüf-Routinen und Auswähllisten von in Aleph hinterlegten Werten (Ctrl+R Listen) auf EB05 in Aleph Test eingespielt und geprüft. Arbeit wird noch fortgesetzt.
- Safari E-Books (Einspielung mit 7000 Tiefen) analysiert und maschinell aufgearbeitet. Einzelne Aufnahmen wurden manuell angepasst und mit Verknüpfungen zu Serien und Normenthalten ergänzt. Die Aufnahmen konnten damit auf ein ausreichendes Level gebracht werden, dass sie im Katalog keine weitere Aufarbeitung durch MDM benötigen. Der Code-Zusatz „-NOA“ wurde daher gelöscht.
- E-Book Aufnahmen wurden im Hinblick auf RDA analysiert und ein Kriterium wurde gesucht, anhand dessen eine E-Book Aufnahme nach RDA erkannt werden kann. Dies ist nötig für künftige E-Book Einspeisungen und den Abgleich mit Aleph.

Erkenntnisse

- Katzen, die mit der Einführung von RDA einfallen, eignen sich hervorragend für das MDM Team. Das bisher erlangte Wissen ist notwendig für bestimmte Arbeiten, wie z.B. Dateneanalysen, Angleichungsmaßnahmen, Expertenbefragungen, sowie den alten Umgang mit GUI. Ebenso ist das Datenflussdiagramm auch für die Einführung der RDA ein Gewinn, da es aufzeigt, welche Systeme Daten aus Aleph beziehen und daher auch von der Umstellung auf RDA betroffen sind.
- Die erstellte Liste mit KofuFo-Beschlüssen zu den einzelnen MARC-Feldern ist eine hervorragende Hilfe für die Konfiguration der Aleph Tabellen. Damit kann sichergestellt werden, dass die Konfigurationen alle KofuFo-Beschlüsse berücksichtigen und auf dem neuesten Stand sind.

https://clouds.ethbib.ethz.ch/nbbit/Dokumente/AP2.1.1_Aenderungen_MARC.pdf

Paas

Platform as a Service

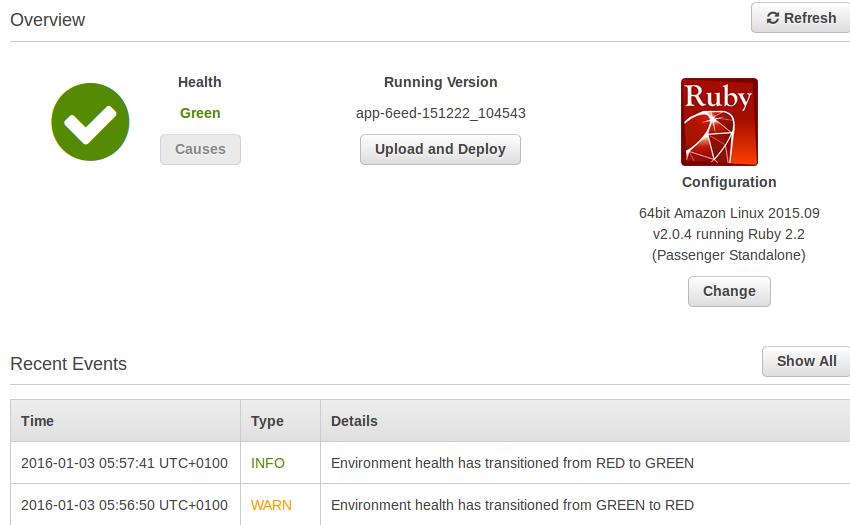


Abbildung 8: Beispiel: AWS: Elastic Beanstalk

Sicherheit

Sicherheit in der IT

Sicherheit

Satz 2. *Der sichere PC steht im Tresor ohne Strom und Netzwerkanschluss.*

Gefahren

- trügerische Sicherheit
- Halbwissen: I hacked 127.0.0.1

trügerische Sicherheit

Anfrage

Guten Tag

Könnten Sie HTTPS für opac.nebis.ch aktivieren?
Ich musste heute das Nebis aus dem Ausland benutzen.

Das Login wurde zwar über <https://login.library.ethz.ch/> geschützt, dann wurde die Verschlüsselung für das

Ausleihen der Bücher wieder abgeschaltet.

Je nach Land kann sogar eine Suche nach einem kritischen Buch zu Problemen führen.

Darum bitte ich Sie, HTTPS auch für das Ausleihen einzuschalten.

Halbwissen

I hacked 127.0.0.1

```
<Elch> Du kennst dich ja voll mit Rechner aus.  
<bitchchecker> halts maul ich hack dich  
<Elch> ok, ich bin ja schon ruhig, nicht dass du uns zeigst was für ein toller Hacker du bist ^^  
<bitchchecker> sag mir deine netzwerk nummer man dann bist du tot  
<Elch> öhm die ist 129.0.0.1  
<Elch> oder war es 127.0.0.1  
<Elch> ja genau die war es: 127.0.0.1 Ich warte dann mal auf einen dollen Hackerangriff  
<bitchchecker> in fünf Minuten ist deine fest platte gelöscht  
<Elch> Da habe ich jetzt aber Angst  
<bitchchecker> halts maul du bist gleich weg  
<bitchchecker> ich hab hier ein program da gebe ich deine ip ein und du bist tot  
<bitchchecker> sag schon mal auf wieder sehen  
<Elch> zu wem?  
<bitchchecker> zu dir mann  
<bitchchecker> buy buy  
<Elch> Ich zitter schon förmlich vor einem so krassen HackOr wie du einer bist  
* bitchchecker (~java@euirc-61a2169c.dip.t-dialin.net) Quit (Ping timeout#)
```

Stay hungry. Stay foolish. Steve Jobs, 2005 in Stanford