

Grundlagen: Informationstechnologie in Bibliotheken

Sven Koesling

ETH-Bibliothek




Herbst 2017

15.12.2017 : Internettechnologien I, Datenbanktechnologien I

- IntT I: Dokumentformen, Skriptsprachen, Ajax, responsive Web
- DBT I: Datenbanktypen, Technologien, Einstieg SQL

- Warum nicht Word, Pages, ...?

1. Dokumentgrößen

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

die Dokumentauszeichnung am Beispiel "Pages"

[†]The 1550 life does not appear to have any viable information associated with it. The deceased row is shown below.

[illegible]

(noch 12 mal soviel...)

der Dokumentinhalt am Beispiel "Pages"

[illegible]

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...).



Die Scheibe.pages

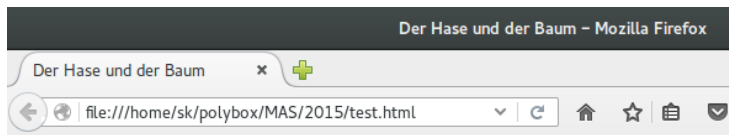


Die Scheibe.pagesX

- Warum nicht Word, Pages, ...?
- html


```
<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 $\frac{1}{4}$ ber ein Feld.

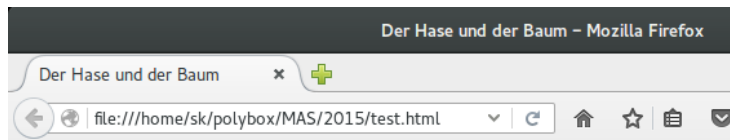
Kapitel2: In der Werkstatt

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.

- Warum nicht Word, Pages, ...?
- html
- pdf

- proprietär, aber offen gelegt

Eigenschaften von pdf

- proprietär, aber offen gelegt
- Papiergrösse, Layout und Inhalt festgelegt

- proprietär, aber offen gelegt
- Papiergrösse, Layout und Inhalt festgelegt
- Text als Text, Bild als Bild enthalten

Wieder die Dokumentgrösse:

Django 2.0: Neue Version des Python-Webframeworks

Oliver Diedrich



Django 2.0 gibt die Unterstützung für Python 2 endgültig auf. Das URL-Routing lässt sich jetzt auch ohne reguläre Ausdrücke aufsetzen.

Ernsthafte Webentwicklung bedeutet heutzutage in der Regel den Einsatz eines Webframeworks, das dem Entwickler viele Routinetätigkeiten abnimmt. In der Python-Welt ist Django, benannt nach dem Jazz-Gitaristen Django Reinhardt, seit zehn Jahren das Webframework der Wahl. Konkurrenten wie Flask konnten bislang nur Nischen besetzen.

Django setzt konsequent das Model-View-Controller-Konzept um, kapselt Datenbankzugriffe über eine objektorientierte Abbildung und generiert automatisch eine Oberfläche zur Verwaltung von Datenbank und Benutzern. Das Framework enthält eine leistungsfähige Template-Sprache und bietet eine flexible URL-Konfiguration.

Die frisch erschienene Version 2.0 ist trotz des großen Versionsprungs weitgehend rückwärtskompatibel zur Vorversion 1.11 – mit einer Ausnahme: Django 2 arbeitet nur noch mit Python 3.4, 3.5 und 3.6 zusammen. Wer noch Python 2.7 nutzt, muss bei Django 1.11 bleiben, das als LTS-Version noch bis April 2020 Sicherheits- und kritische Bugfixes erhalten soll. Django 1.10 hat mit dem Erscheinen der neuen Version das Ende des Supports erreicht.

als Bild: 178.8 kB

als PDF: 67.7 kB

Fazit: Welche Form wofür?

Textverarbeitung: Alles zum Weiterverarbeiten

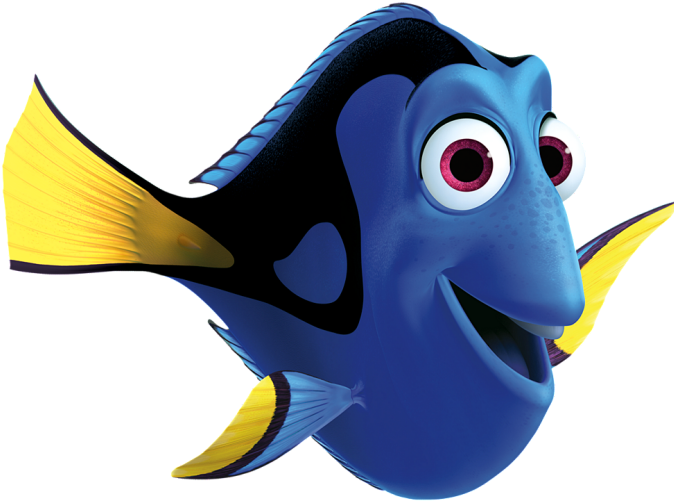
html: Inhalt geht über Form

PDF: Form soll erhalten bleiben

Theorem

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

http is stateless



...some web applications may have to track the user's progress from page to page...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)

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

JS



Das erste Programm

Number of Operations	Nature of Operations	Variables acted upon	Variables receiving results	Indication of change in the value on any Variable	Statements of Results	State												Working Variables				Result Variables																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																							
						$1V_1$	$1V_2$	$1V_3$	$1V_4$	$1V_5$	$1V_6$	$1V_7$	$1V_8$	$1V_9$	$1V_{10}$	$1V_{11}$	$1V_{12}$	$1V_{13}$	$1V_{14}$	$1V_{15}$	$1V_{16}$	$1V_{17}$	$1V_{18}$	$1V_{19}$	$1V_{20}$																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																				
○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○

Die Aufgabe lautet, ein Programm zu schreiben, das die Zahlen von 1 – 100 hoch zählt und

- bei Zahlen, die durch **drei** teilbar sind, „Fizz“ ausgibt,
- bei Zahlen, die durch **fünf** teilbar sind, „Buzz“ ausgib und
- bei Zahlen, die durch **beides** teilbar sind, „FizzBuzz“ ausgibt.

Ansatz in Perl

```
#!/usr/bin/perl
for my $zahl (1 .. 100){
    if ($zahl % 15 == 0) {
        print "Fizz Buzz\n";
    }
    elsif ($zahl % 5 == 0) {
        print "Buzz\n";
    }
    elsif ($zahl % 3 == 0) {
        print "Fizz\n";
    }
    else {
        print "$zahl\n";
    }
}
```

```
#!/usr/bin/ruby
(1..100).each do |z|
  if z.modulo(15) == 0
    puts "FizzBuzz :-)"
  elsif z.modulo(5) == 0
    puts "Buzz"
  elsif z.modulo(3) == 0
    puts "Fizz"
  else
    puts z
  end
end
```

Ein Programm

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

```
Dir.foreach('.') do |item|
  next unless item =~ /\.tgz$/ or item =~ /tar.gz$/
  puts "===== \ndurchsuche #{item}:"
  cmd = "tar tzf #{item}"
  ergebnis = %x[ #{cmd}]
  ergebnis.each_line do |file|
    suchstring.each do |s|
      if file =~ /#{s}/
        puts "#{file.chomp} gefunden."
        print "Soll die Datei extrahiert werden? [j/N] : "
        antwort = STDIN.gets
        if antwort =~ /[jyY]/
          puts "OK, ich extrahiere #{file.chomp} aus #{item}."
          if File.exist?(file.chomp)
            ts = Time.now.to_i
            %x[/bin/tar --transform 's/${ts}-/' -xzf #{item} #{file}]
          else
            %x[/bin/tar xzf #{item} #{file}]
          end
        end
      end
    end
  end
end
```

Interaktion mit dem User durch:

Asynchronous **J**avaScript **A**nd **X**ML

Wo wird was ausgeführt?

?

DOM

was kann wo manipuliert werden?

[Neueinschreibungen](#)[Nicht freigeschaltet](#)[alle Benutzer \(40\)](#)[Tagesgäste](#)[Hilfe](#)[Meine Daten](#)

Sie sind als Administrator eingeloggt

[Suchen](#)

Nachname, Vorname	Geb.	E-Mail	eingeschr.	Kommentar
Test, Lars (Kein Entitlement)	27.06.1978	📧 dalars27@gmail.com	5.9.2017	
Lüttringhaus, Rahel (Kein Entitlement)	08.03.1982	📧 rahel.luettringhaus@library.ethz.ch	4.9.2017	
Garkeiner, Niemand (Kein Entitlement)	01.01.2000	📧 mcvsvenster@gmail.com	10.1.2017	

DOM und Klassen

Schauen wir uns das mal im Quelltext an.

ein Benutzer mit verifizierter EMail-Adresse:

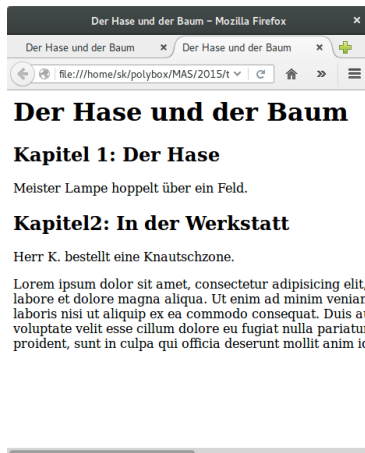
```
<div class="col-md-3">  
  <a href="mailto:rahel.luettringhaus@library.ethz.ch" class="verified"><span class="glyphicon glyphicon-ok-sign"  
</div>
```

...und einer mit einer unbekannten EMail-Adresse:

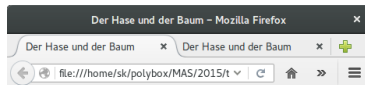
```
<div class="col-md-3">  
  <span class="unknown"><span class="glyphicon glyphicon-question-sign"  
</div>
```


Responsive Web

So soll es nicht sein!



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 adipiscing 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.

Warum Datenbanken?

Warum Datenbanken?

- geringer Speicherplatzbedarf

Warum Datenbanken?

- geringer Speicherplatzbedarf
- gleichzeitiger Zugriff durch viele Nutzer

Welche Typen von Datenbanken gibt es?

Welche Typen von Datenbanken gibt es?

- K/V– Stores

Welche Typen von Datenbanken gibt es?

- K/V– Stores
- relationale Datenbanken

Welche Typen von Datenbanken gibt es?

- K/V– Stores
- relationale Datenbanken
- Spaltenorientierte Datenbanken

Welche Typen von Datenbanken gibt es?

- K/V– Stores
- relationale Datenbanken
- Spaltenorientierte Datenbanken
- Dokumentorientierte Datenbanken

Welche Typen von Datenbanken gibt es?

- K/V– Stores
- relationale Datenbanken
- Spaltenorientierte Datenbanken
- Dokumentorientierte Datenbanken
- Graphendatenbanken

K/V– Stores sind — wie der Name schon sagt — schlichte Schlüssel / Wert– Speicher. Sie sind bei minimalem Speicherplatzbedarf sehr schnell, was sie für das Caching von Werten prädestiniert.

relationale Datenbanken

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

relationale Datenbanken

Tabelle der Strassen

id	Strasse
1	Hauptstrasse
2	Nebenstrasse
3	Seitenstrasse

relationale Datenbanken

Tabelle der Städte

id	Stadt
1	Zürich
2	Basel
3	Bern

relationale Datenbanken

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

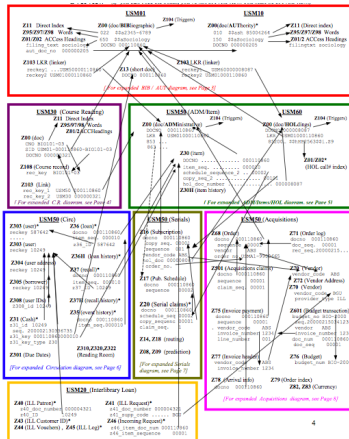
Eine relationale Datenbank dahingehend zu optimieren, dass es möglichst wenig Redundanzen gibt, nennt man „normalisieren“. Es gibt fünf Normalformen.

relationale Datenbanken

Normalisierung: 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
```

Relationen in der DB von Aleph



relationale Datenbanken

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'  
;
```

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

relationale Datenbanken

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]*

relationale Datenbanken

Quasi-Monopole



Abbildung: <http://houseofbrick.com/the-oracle-parking-garage/>

```
{ "_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

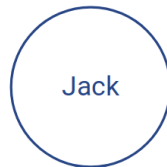
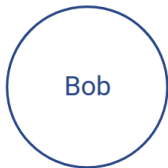
```
{ "_id": 3,  
  "Vorname": "Martha",  
  "Nachname": "Graham",  
  "Beruf": ["Tänzerin", "Choreographin"]  
}
```

und noch ein weiteres Dokument in derselben collection

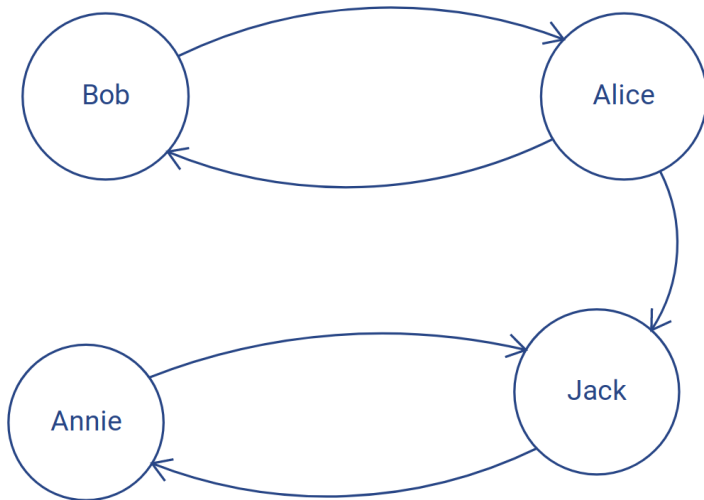
```
{ "_id": 4,  
  "Vorname": "Niemand",  
  "Nachname": "Nixda",  
  "email": "nn@none.org"  
  "Hund": {  
    "Name": "Nero",  
    "Geschlecht": "male"  
  }  
}
```

Graphendatenbanken sind quasi Knoten und Kanten mit Attributen.

ein paar Knoten



Knoten und Beziehungen



Knoten, Beziehungen und Attribute

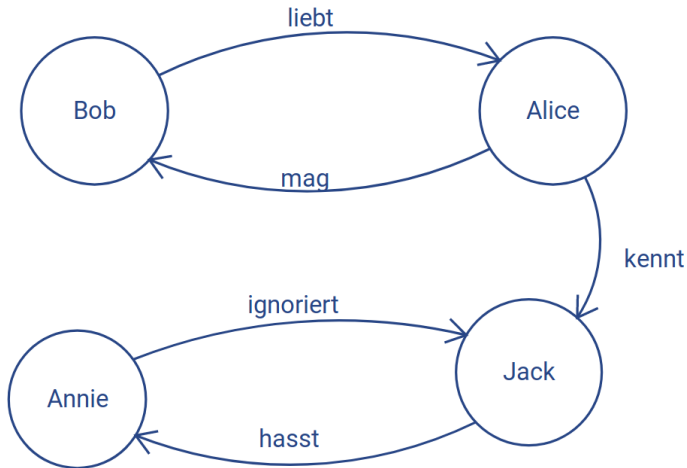


Abbildung: Suche nach "Landquart" in einem aktuellen Bibliothekskatalog (420.000 Dokumente): 1072 Treffer – ohne Facetten: completed in 2845ms



[Ihr Bibliothekskonto](#)
Suche

[Neustart](#)
Ergebnisliste

[Hilfe](#)
Suchverlauf

[Ihre Literaturliste](#)

[Drucken](#)

[Teilkataloge BGR](#)
[Andere Kataloge](#)

[Auswahl anzeigen
In Ihre Literaturliste](#)
[Alle auswählen
Versenden](#)
[Auswahl entfernen
Speichern](#)
[Untermenge](#)
[Suche modifizieren](#)
[Filtern](#)

Ergebnisse

für "Stichwort= Landquart"; Sortiert nach: Jahr, dann Autor
Sätze 1 - 10 von 1072

[Gehe zu #](#)
[Vorige Seite](#)
[Nächste Seite](#)

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

[Vorige Seite](#)
[Nächste Seite](#)

Abbildung: 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

Albula Berrina Hinterrhein Imboden Inn **Landquart** Maloja Moesa Plessur Prättigau / Davos Surselva

Feedback

Ergebnis für "**Landquart**"
(1700 Treffer)

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

Kreis

Fünf Dörfer
Maienfeld

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

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

Autoren (47)

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

"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](#)

Urheber (31)

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

"Ausgelöffelt" - "Hirschen" in Maienfeld

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

"Bock auf Rock!"

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

Schlagworte (26)

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

"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"