Einsteiger-Tutorial zu Korpusrecherchen mit WaCkY

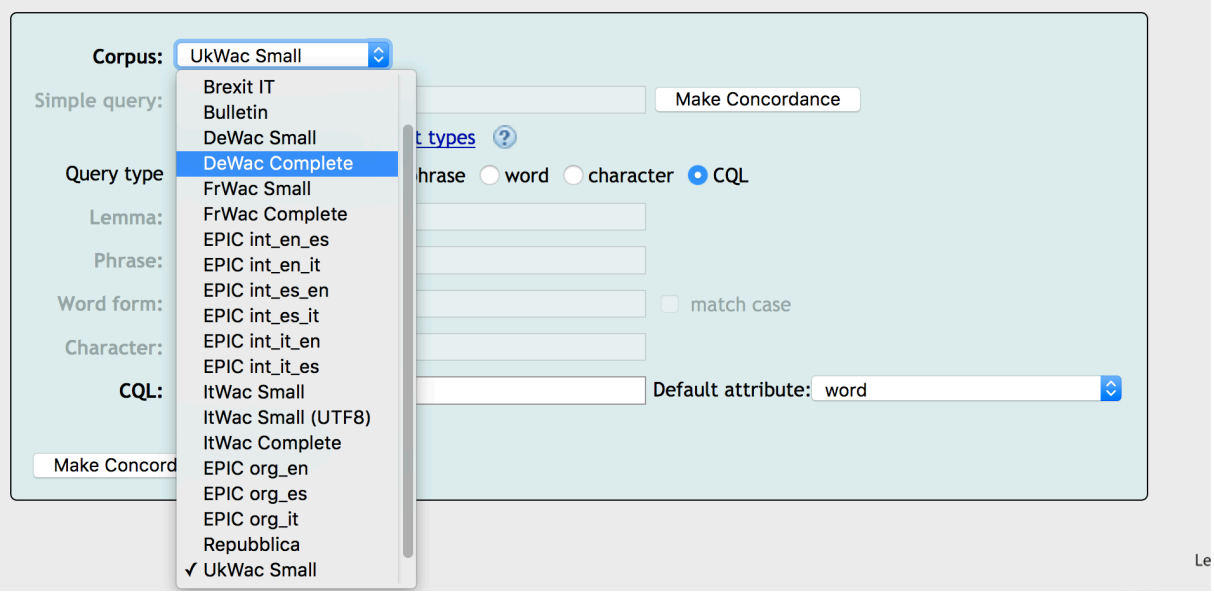
Um Sprachdaten aus dem Internet zu gewinnen, waren Linguistinnen lange auf konventionelle Suchmaschinen angewiesen. Dies hatte den Nachteil, dass quantitative Analysen aufgrund der stetig schwankenden Grundgesamtheit (und auch aufgrund der unterschiedlichen Suchalgorithmen) praktisch unmöglich waren. Mit Internetkorpora wie COW und WaCkY ändert sich dies derzeit. Auch wenn sich diese Korpora derzeit noch vor allem durch ihre schiere Größe auszeichnen und Metainformationen, die gerade aus diachroner Sicht wünschenswert wären, noch fehlen (z.B. Erstellungsjahr / Jahr der letzten Änderungen an den jeweiligen Seiten, was „mikro-diachrone" Analysen möglich machen würde), handelt es sich hier um wertvolle neue Ressourcen zur Erforschung jüngster Sprach(wandel)phänomene.
Dieses Tutorial bezieht sich nur auf WaCkY, das derzeit (Stand 25.02.2020) über https://corpora.dipintra.it/[1] (Button „Public") direkt und ohne Anmeldung durchsucht werden kann.

Angenommen, wir wollen mit Hilfe des deWaC-Korpus den Gebrauch des Zweitglieds (Suffixoids?) *-papst* untersuchen, z.B. *Literaturpapst, Splatterpapst*.

Hinweis: Bei der Arbeit mit den exportierten Dateien verwendet dieses Tutorial Notepad++. Dieses kostenlose Programm ist hier erhältlich: http://notepad-plus-plus.org/ Es ist nur für Windows verfügbar. Für Mac bietet z.B. BBEdit (https://www.barebones.com/products/bbedit/) einen ähnlichen Funktionsumfang. Dabei handelt es sich um ein kommerzielles Produkt, das aber mit eingeschränktem und für uns völlig ausreichendem Funktionsumfang unbegrenzt kostenlos genutzt werden kann. Für Linux gibt es z.B. Notepadqq.

1. Ein Korpus auswählen

Im NoSketchEngine-Interface müssen Sie zunächst ein Korpus auswählen. Da wir mit deutschsprachigen Seiten arbeiten wollen, nehmen wir deWaC:



[1] Hinweis: Dieses Tutorial wurde ursprünglich für eine andere NoSketchEngine-Instanz geschrieben (http://nl.ijs.si/noske/wacs.cgi/first_form), die jedoch einige Bugs aufweist. Deshalb sind minimale Abweichungen bei den Screenshots etc. möglich.

Mit einem Klick auf **Query Type** lassen sich unterschiedliche Anfragetypen auswählen. Diese sind weitgehend selbsterklärend: Wenn Sie mit einer „simple query“ z.B. *Papst* suchen, werden alle Treffer für *Papst* angezeigt, nicht aber z.B. für *Papstes, Päpste* usw. Wenn Sie *Papst* in der Lemmasuche eingeben, werden auch die unterschiedlichen Flexionsformen gefunden. Unter „Phrase“ können Sie nach Phrasen wie *die Eier vom Papst* suchen (immerhin ein Treffer). Mit CQL können Sie in der **Corpus Query Language** suchen. Diese Option ist für unsere Fragestellung attraktiv, deshalb schauen wir sie uns genauer an.

2. Corpus Query Language

Die Sprache des zur IMS Corpus Workbench gehörenden *corpus query processor* (CQP) ist ausgesprochen vielseitig und lässt sich auch für recht komplexe Suchanfragen verwenden. Zugleich ist die grundlegende Syntax sehr einfach zu lernen und intuitiv nachvollziehbar. Hier werden nur die „Basics“ behandelt, für Näheres sei z.B. auf Stefan Everts Tutorial verwiesen: *cwb.sourceforge.net/files/**CQP_Tutorial**.pdf*

CQP funktioniert nach dem Prinzip: **ein** Token in **einer** eckigen Klammer, nach dem Muster `[attribut="wert"]`, wobei attribut z.B. die Wortform sein kann (`word`), das Lemma (`lemma`) oder die Wortart (`pos`, für part of speech). Unterschiedliche Attribute, die sich auf dasselbe Token beziehen, werden mit & verbunden.

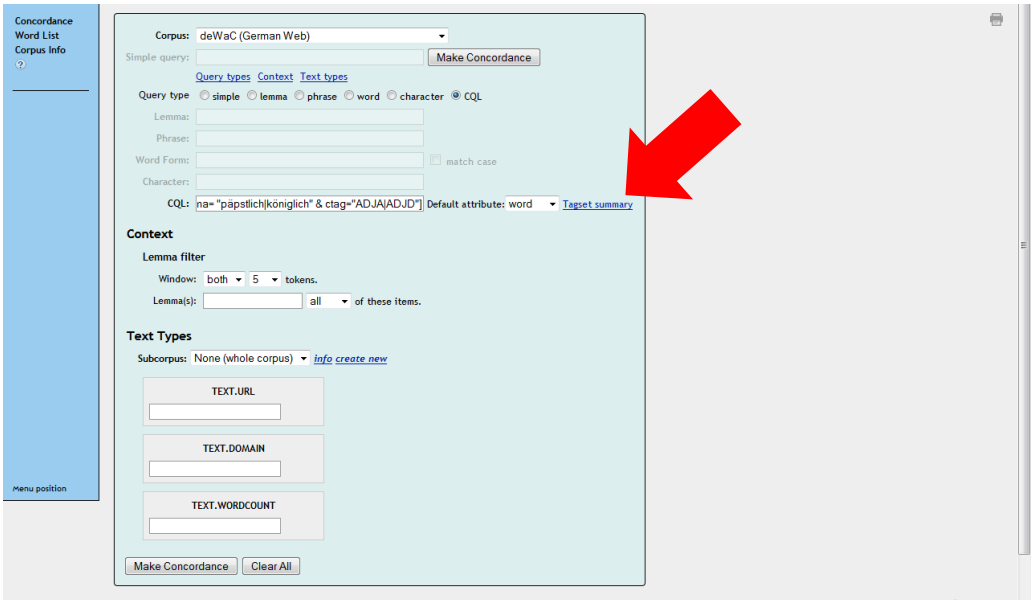
Andere Korpora - andere Sitten!
Damit die Suche nicht zu einfach wird, unterscheiden sich die Attribute teilweise von Korpus zu Korpus. Während „pos“ der verbreitetste Attributname für Wortarten ist, müssen Sie in deWaC mit „ctag“ arbeiten. Auch die Werte unterscheiden sich: Zum Beispiel sind Adjektive in einigen Korpora „ADJ“, in anderen „JJ“. Das hängt davon ab, welches Tagset verwendet wurde. Für Wortarten ist zum Beispiel das Stuttgart-Tübingen Tagset (STTS) sehr verbreitet, aber z.B. auch das CLAWS-Tagset.

```
[lemma= "päpstlich" & ctag="ADJA"]
```

findet z.B. alle Belege, in denen *päpstlich* als attributives Adjektiv gebraucht wird.
Neben & sollten Sie noch | kennen, was „oder“ bedeutet.

```
[lemma= "päpstlich|königlich" & ctag="ADJA|ADJD"]
```

findet alle Belege für *königlich* oder *päpstlich*, die als attributives oder prädikatives Adjektiv annotiert sind.
Nun fragen Sie sich bestimmt, wo Sie die unterschiedlichen Werte für Wortartenkategorien nachschlagen können. Hier können Sie das Tagset benutzen, das auf der Query-Seite verlinkt ist.



Für unsere Beispielsuche sind die POS-Tags jedoch irrelevant, da wir bei *-papst* ohnehin davon ausgehen können, nur Substantive zu finden.
Wie finden wir nun aber Fälle, in denen vor dem *-papst* noch etwas anderes kommt? Hier müssen wir mit **Wildcards** arbeiten, also Zeichen, die für ein beliebiges anderes Zeichen stehen. Sie kennen sicherlich den Asterisk (*) als Wildcard: In COSMAS z.B. müssten Sie mit **papst* nach dem *Literaturpapst* oder dem *Reisepapst* suchen. In CQL übernimmt der **Punkt** diese Funktion. Der Asterisk fungiert dagegen als **Wiederholungsoperator**. Wiederholungsoperatoren finden Fälle, in denen das damit modifizierte Zeichen wiederholt wird. Zum Beispiel fände *e* e, ee, eee* usw. usw.

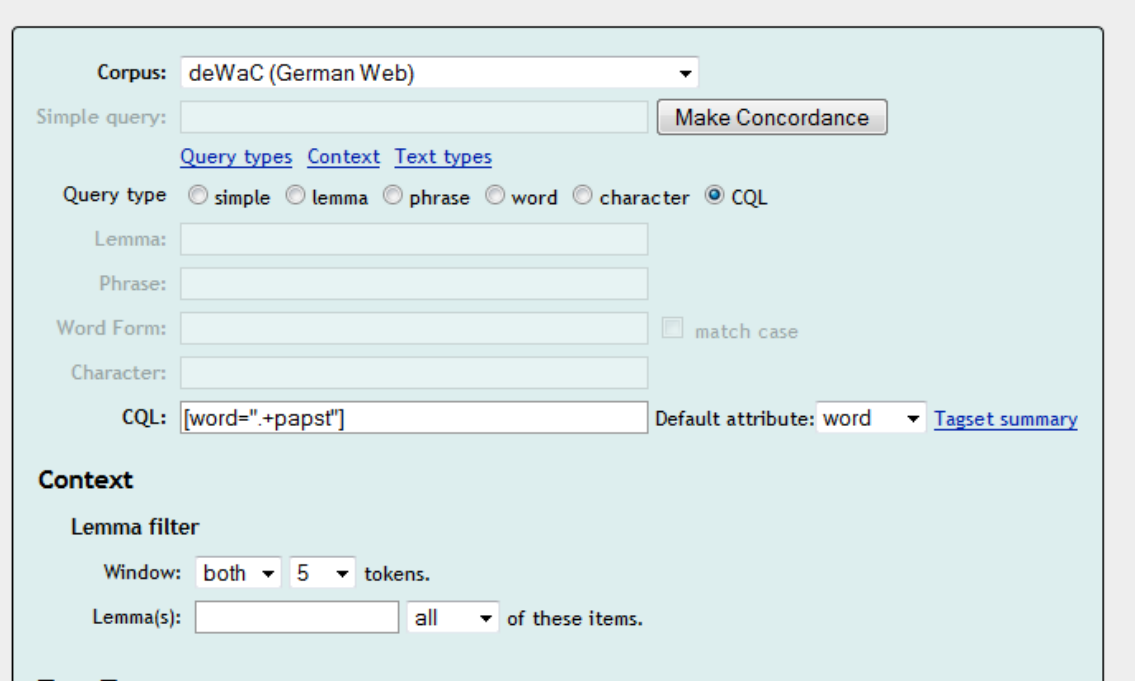
Wiederholungsoperatoren:

*	mindestens 0-mal
+	mindestens 1-mal
?	mindestens 0-mal, höchstens 1-mal
{n}	genau n-mal (z.B. {2}: genau 2-mal)
{n,m}	mindestens n-mal, höchstens m-mal (z.B. {1,5}: mind. 1-mal, höchstens 5-mal)

Durch die Kombination von Wildcard und Wiederholungsoperatoren können wir spezifizieren, dass vor dem *-papst* noch mindestens ein anderer Buchstabe stehen soll.

[word=".+papst"]

Diese Anfrage findet den *Literaturpapst*, nicht aber den *Literatur-Papst.* Bevor Sie weiterlesen: Wie müssen wir die Anfrage modifizieren, um auch den *Literatur-Papst* zu finden?



Die Antwort:
`[word=".+papst|.+-Papst"]`

Hinweis zu Escape-Zeichen

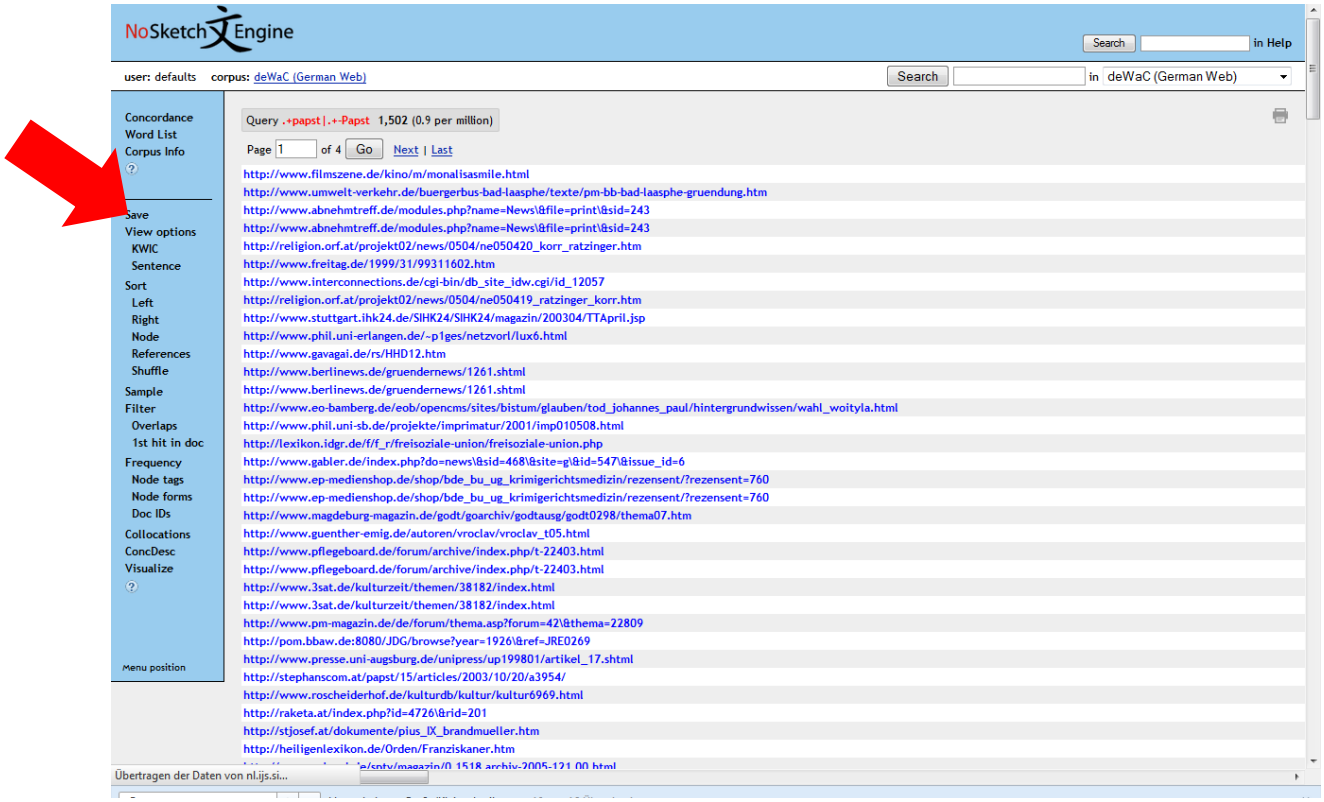
Sie sehen, dass wir in unserer Suchanfrage auch nach einem Interpunktionszeichen suchen, nämlich -. In diesem Fall geht das problemlos. Viele Satzzeichen haben jedoch in CQP eine andere **Funktion**: Der Punkt zum Beispiel wird, wie wir gesehen haben, als Wildcard verwendet. Die Suche nach `[word="einfach"] [word="weil. "]` findet daher nicht etwa alle Belege, bei denen auf *einfach weil* ein Punkt folgt, sondern alle Belege, in denen auf das *weil* noch ein weiteres Zeichen folgt, z.B. *einfach weils Spaß macht*. Hier müssen Sie mit dem Slash (\) als sog. Escape-Zeichen arbeiten:

`[word="einfach"] [word="weil\."]`

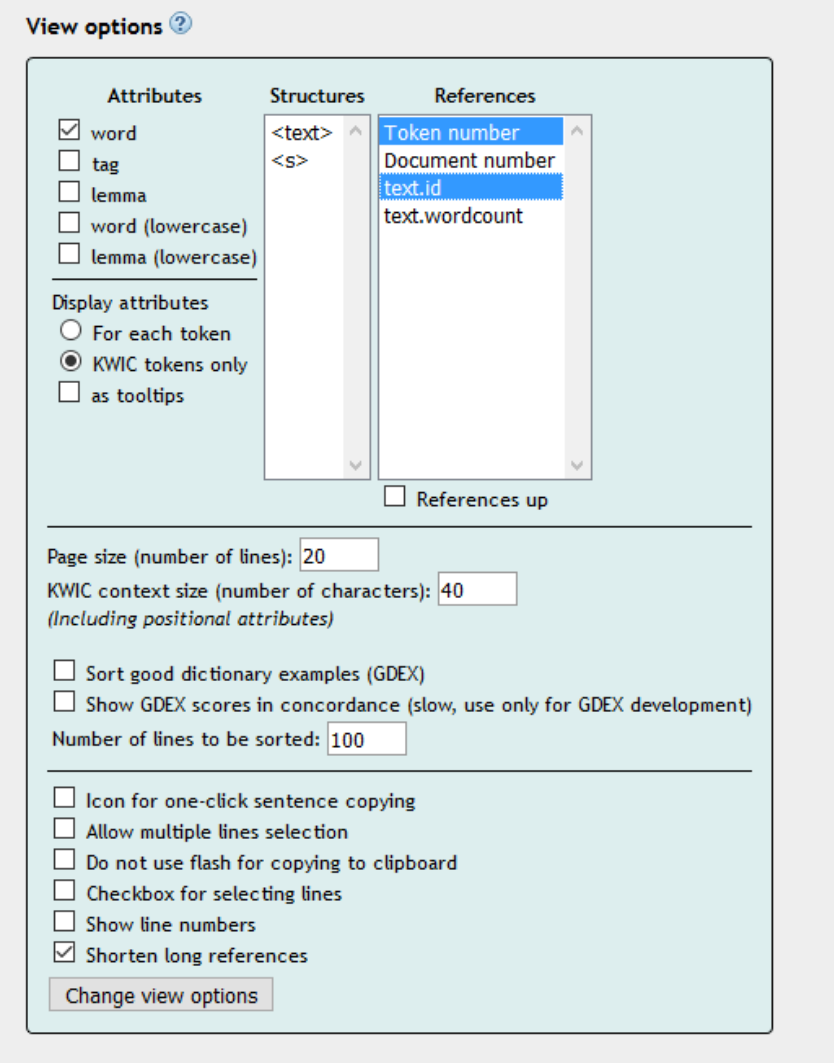
(Hinweis: *einfach weil.* ist in deWaC leider nicht belegt, aber wenn Sie mir nicht glauben, probieren Sie stattdessen `[word="weil\."])`

3. Export

Die Suche nach `[word=".+papst|.+-Papst"]` bringt uns, nach kurzer Rechenzeit, zur **Konkordanz**, d.h. zur Belegsammlung. Diese können wir auch **exportieren:** Mit einem Klick auf „Save“, das sich in der Spalte links verbirgt.



Vor dem Export können wir mit Klick auf **ViewOptions** (ebenfalls in der linken Spalte) noch konfigurieren, welche Informationen wir in der Darstellung der Konkordanz (sowohl online als auch letztlich in der Exportdatei) sehen wollen:



Wenn, anders als hier im Screenshot, bei **Attributes** alle Häkchen gesetzt sind, sieht die Konkordanz so aus:

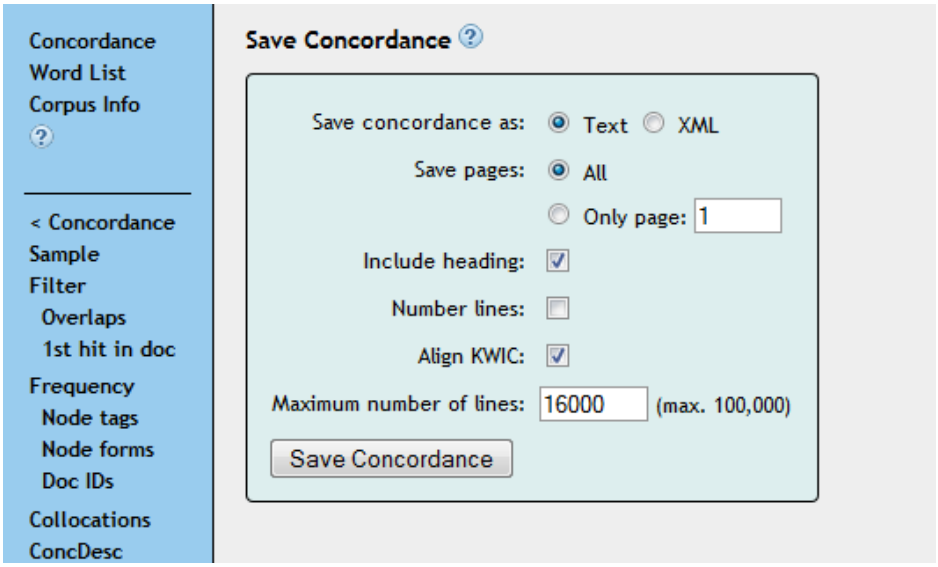
```
Die Heerschar, angeführt vom allseits beliebten Kritikerpapst
/Kritikerpapst/NN/Nc/So  Marcel R. R. , glaubte , in " Sab-
baths Theater " keine Geschichte finden zu können .
```

Was wie folgt zu entschlüsseln ist:

Kritikerpapst	Wortform
Kritikerpapst	Lemma
NN	Nomen
Nc	common noun (=Appellativ, kein Eigenname)
So	nochmal common noun, diesmal nach slowenischer Annotation.

Entscheiden Sie, welche dieser Tags Sie für Ihre Fragestellung benötigen. Für unsere Beispielsuche entfernen wir alle Häkchen außer natürlich das bei **word**, damit wir in unserer Konkordanz nicht mit einem „Rattenschwanz“ an Tags kämpfen müssen. In der Spalte „References“ können Sie außerdem noch auswählen, welche Metadaten exportiert werden sollen.

Nun klicken wir endlich auf **Save,** wo wir zunächst die Optionen für den Export konfigurieren können:



Insbesondere müssen wir die **Maximum number of lines** erhöhen, die per Default auf 1000 gesetzt ist, was bedeutet, dass nur die ersten 1000 Belege exportiert würden. Für den *-papst* gibt es etwas über 1500 Belege, also nehmen wir hier 1600. (Oder gleich 16000 wie im Screenshot...)
Außerdem wählen wir statt Text **XML**. Und zwar deshalb, weil das Keyword (also die Wörter, nach denen wir suchen, z.B. *Reisepapst, Fitnesspapst*) in der .txt-Variante mit < und > umschlossen ist, aber auch in den Belegen selbst z.T. < und > vorkommen. Die xml-Variante sieht für Ungeübte vielleicht zunächst etwas furchteinflößend aus, aber im Grunde ist die Vorgehensweise dieselbe, die wir auch bei der .txt-Version wählen würden – nur zuverlässiger.

Im weiteren Verlauf des Tutorials geht es darum, die Dateien in ein Spreadsheet für Tabellenkalkulationsprogramme zu überführen. Wenn Sie das nicht manuell machen möchten, können Sie mein R-Paket *concordances* verwenden (das es noch nicht gab, als ich das Tutorial geschrieben habe). Das Paket mit Installationsanleitung finden Sie hier:
https://github.com/hartmast/concordances
Eine deutschsprachige Schnellanleitung finden Sie unter
https://tinyurl.com/concordances-schnellanleitung
Um WACKY-Konkordanzen einzulesen, benutzen Sie *getWACKY* und geben Sie als Argument den Pfad zur heruntergeladenen XML-Datei ein, also z.B.

```
getWACKY("/Users/stefanhartmann/Downloads/dewac_concordance.xml")
```

Nachdem wir die xml-Datei **gespeichert** haben, **öffnen** wir sie **mit Notepad++** (bzw. unter Mac mit BBEdit oder einem anderen Texteditor mit ähnlichem Funktionsumfang).

4. XML > Excel

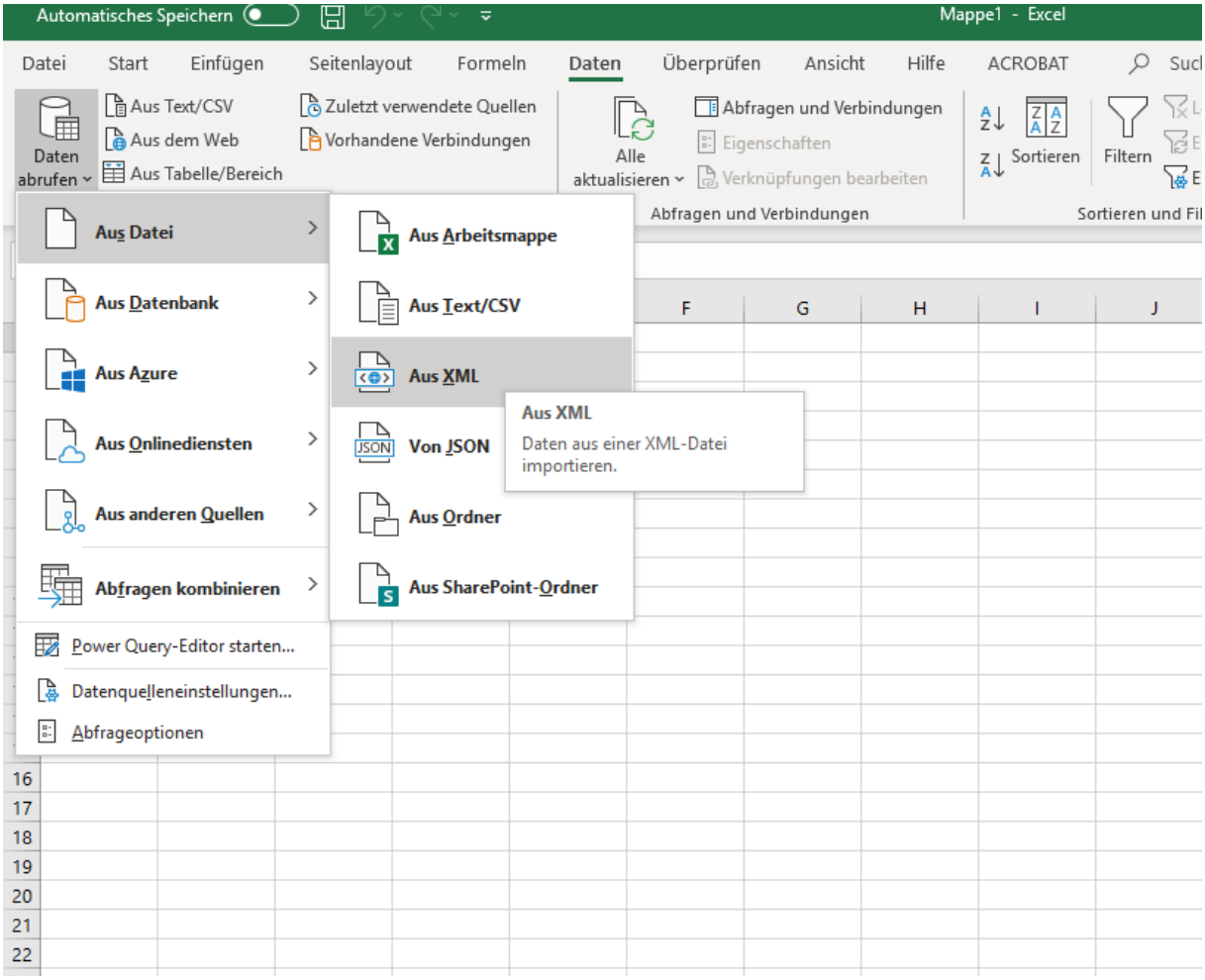
Nun wollen wir die exportierten Daten in ein Tabellenkalkulationsprogramm übertragen. Hierfür gibt es mehrere Möglichkeiten:

1. Der automatische Weg

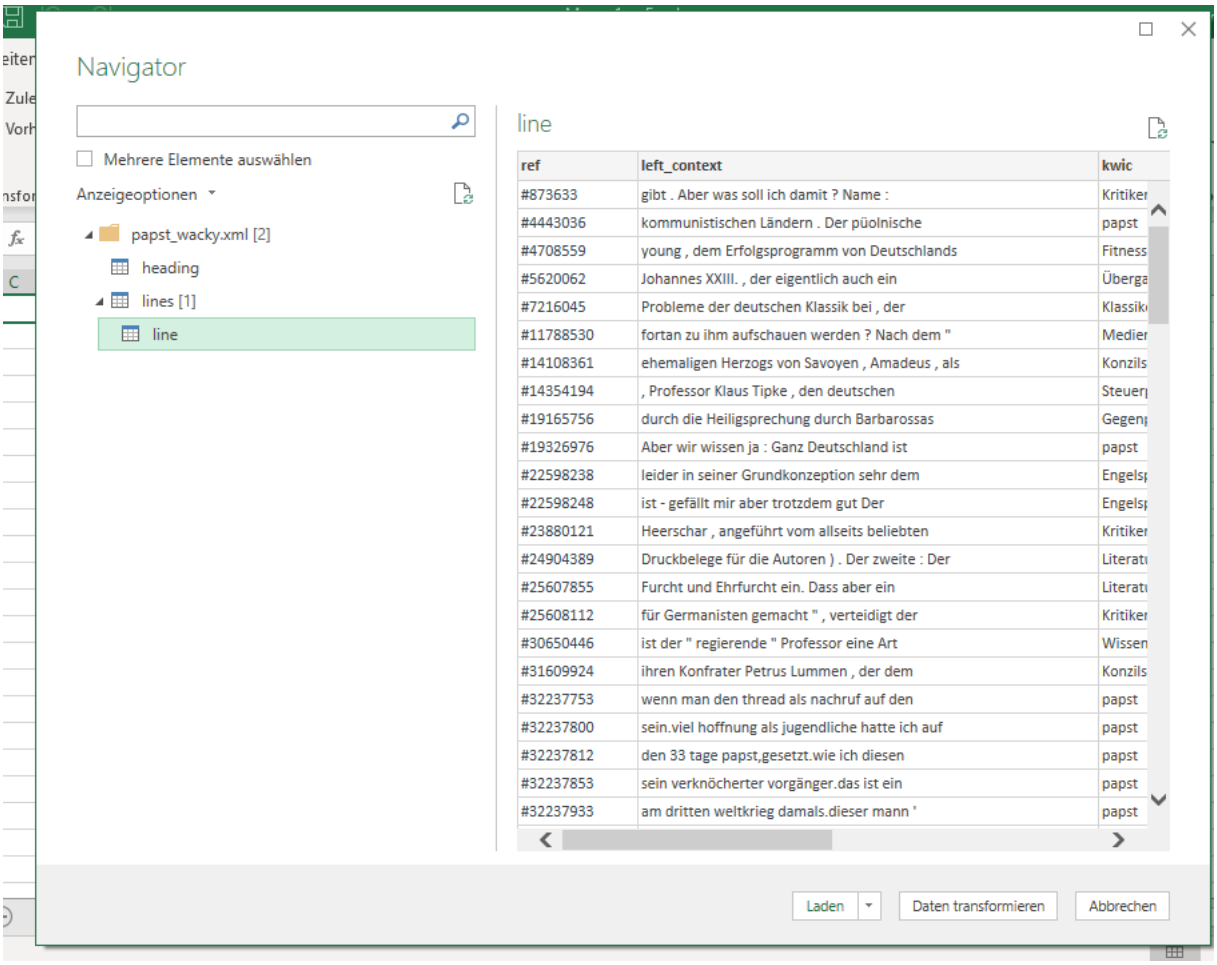
Excel für Windows verfügt über eine XML-Importfunktion. In Excel für Mac ist diese Funktion leider (noch) nicht verfügbar. Falls Sie einen Mac oder Linux nutzen, können Sie stattdessen das kostenlose Pendant LibreOffice Calc zum Dateiimport nutzen. Falls Sie lieber mit Excel arbeiten, können Sie die Datei nach dem Datenimport in Calc auch als .xlsx-Datei speichern, in Excel öffnen und dort weiterarbeiten.

Der Dateiimport in Excel für Windows und LibreOffice Calc funktioniert sehr ähnlich. Zunächst zeige ich, wie er in Excel funktioniert:

Im Reiter „Daten“ wählen Sie unter „Daten abrufen“ die Option Aus Datei > Aus XML.

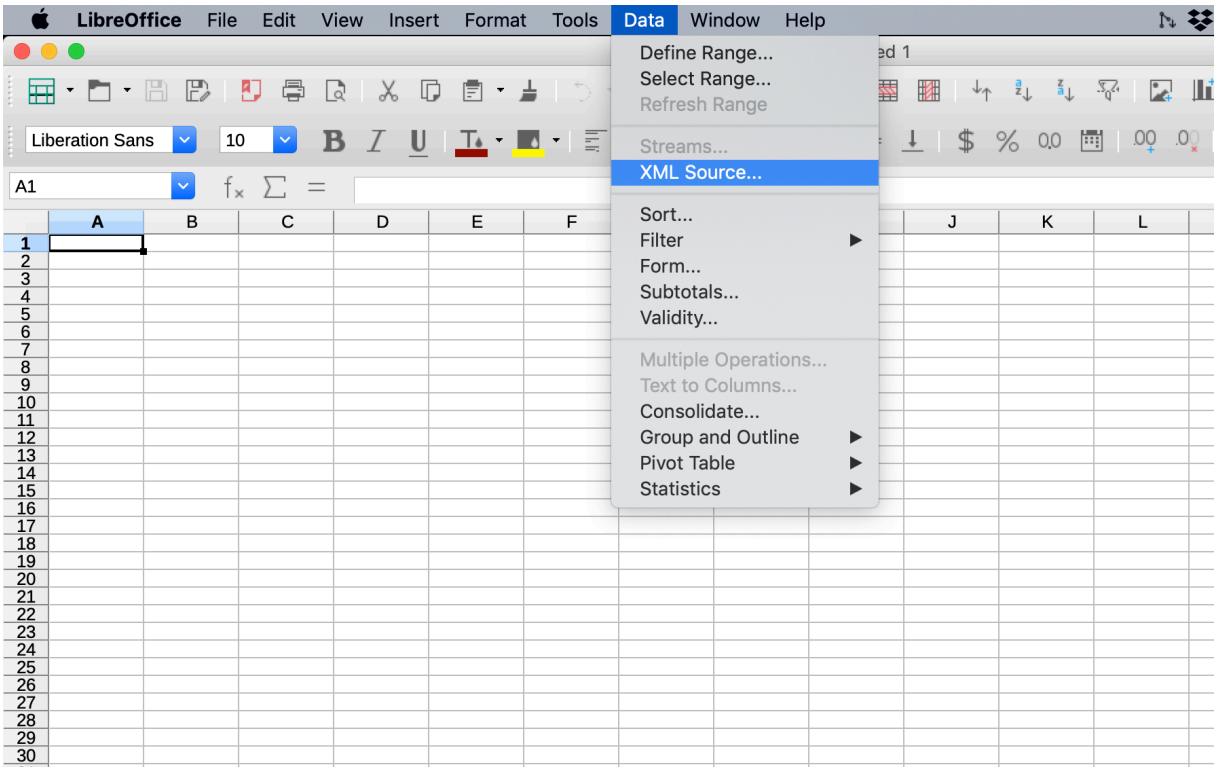


Daraufhin öffnet sich ein Fenster, in dem Sie die XML-Datei auswählen können. Anschließend öffnet sich ein Fenster, in dem wir links die hierarchische Struktur der XML-Datei sehen. Rechts gibt es ein Vorschaufenster, das zunächst noch leer ist – das liegt einfach daran, dass die hierarchische Struktur der XML-Datei nicht in einer zweidimensionalen Tabelle dargestellt werden kann. Das ist aber nicht schlimm, denn wir benötigen ohnehin nur das niedrigste Element in der Hierarchie, das die eigentliche Konkordanz enthält. Es ist der Knoten „line“ – wenn wir diesen anwählen, sehen wir auch die Konkordanz:

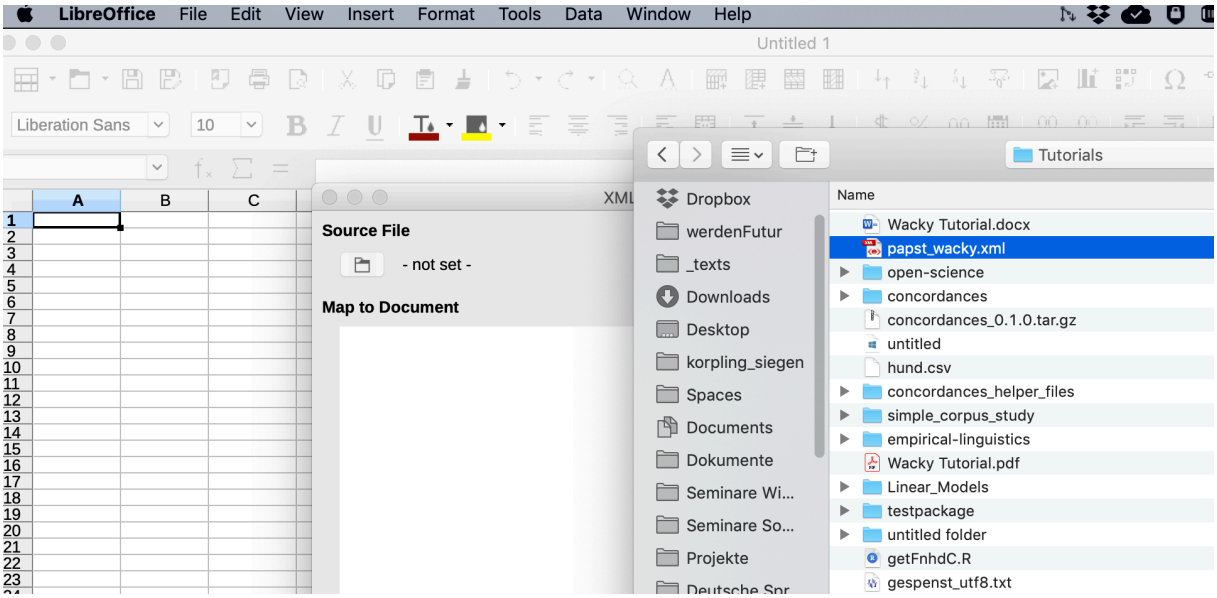


Klicken wir nun auf den Button „Laden“, werden die Daten eingelesen und gleich als Tabelle formatiert.

In LibreOfficeCalc ist der XML-Import über Data > XML Source möglich:

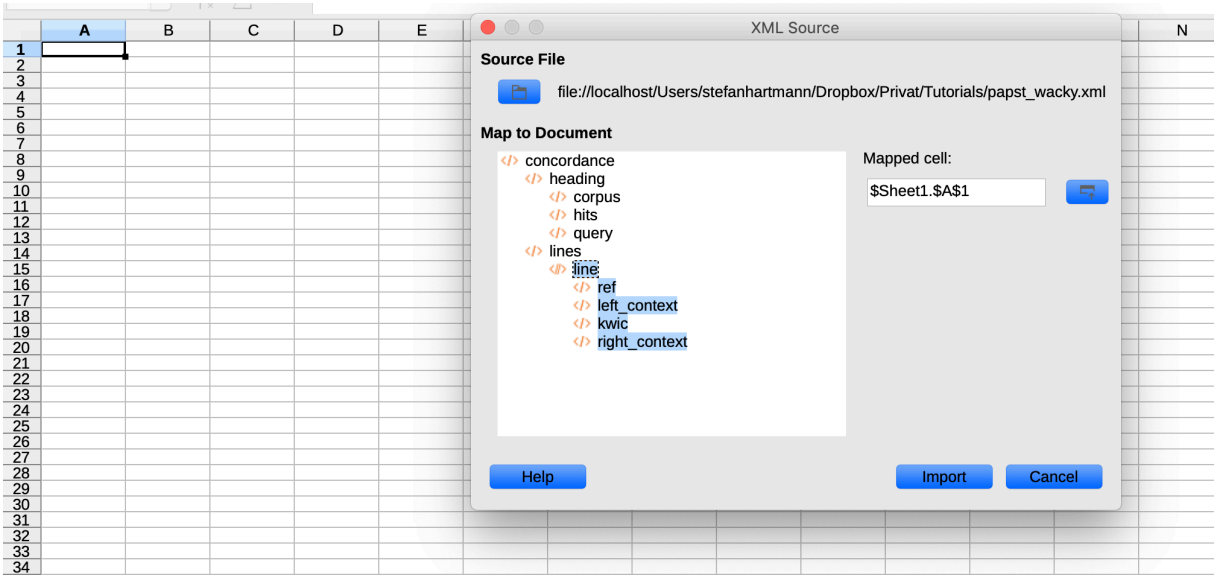


Es öffnet sich ein Fenster, in dem Sie mit Klick auf das kleine Icon unter „Source File“ öffnen Sie einen Dialog, in dem Sie die Datei auswählen können.



Als nächstes sehen Sie die hierarchische Struktur des XML-Dokuments. Sie werden bemerken, dass der Button „Import“ zunächst ausgegraut ist. Das liegt daran, dass wir zuerst eine Zelle auswählen müssen, auf die wir die Daten „mappen“, d.h. in die wir die Daten importieren wollen. Dafür klicken Sie auf das Icon rechts neben dem leeren Feld unter „Mapped Cell“ und wählen z.B. die Zelle A1. Im Diagramm links, das die hierachische Struktur des Dokuments abbildet, müssen wir auswählen, welchen Teil des Dokuments wir importieren möchten. Hier wollen wir nur die eigentliche Konkordanz, nicht die Header-Daten, die ebenfalls im

Dokument enthalten sind. Wir wählen daher den letzten „Knoten“ in der XML-Struktur aus, *line* (das die vier Tabellenspalten als „Töchter“ hat) und klicken auf „Import“ – schon haben wir unsere Konkordanz importiert.



Sollte das nicht funktionieren oder sollten Sie aus irgendwelchen anderen Gründen manuell arbeiten wollen, finden Sie im Folgenden die Anleitung zum manuellen Import, die ich geschrieben habe, bevor ich die Möglichkeit des XML-Imports entdeckte...

2. Der manuelle Weg

Um die Daten in ein Tabellenkalkulationsprogramm zu übertragen, müssen wir sie noch etwas bearbeiten. Es empfiehlt sich, ein neues Textdokument anzulegen und alles unterhalb des Headers (erkennbar an „/heading“) in dieses neue Dokument zu copy&pasten.

Durch die farbliche Hervorhebung in Notepad sehen Sie schon ein Muster: Jede einzelne Belegzeile ist umschlossen mit

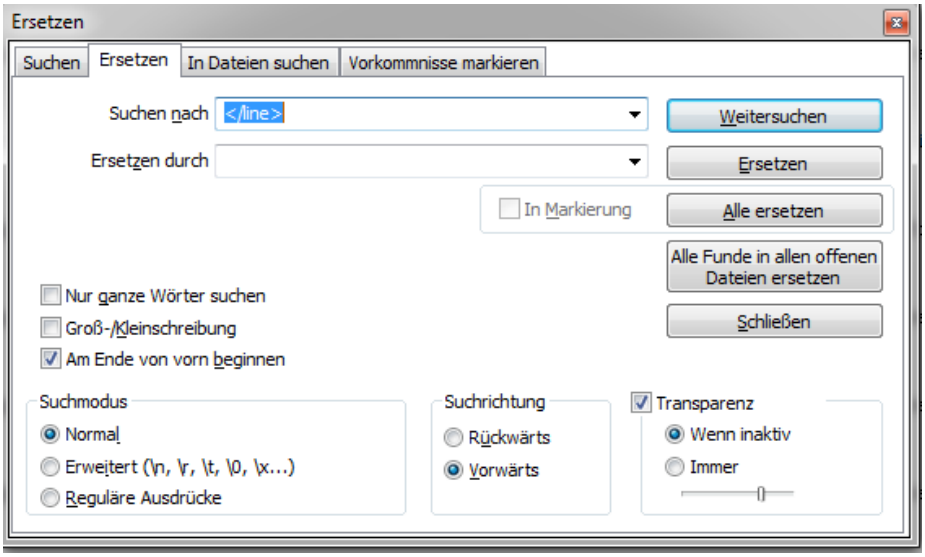
<line>
</line>

Innerhalb der einzelnen Zeilen setzt sich jeder Beleg zusammen aus:

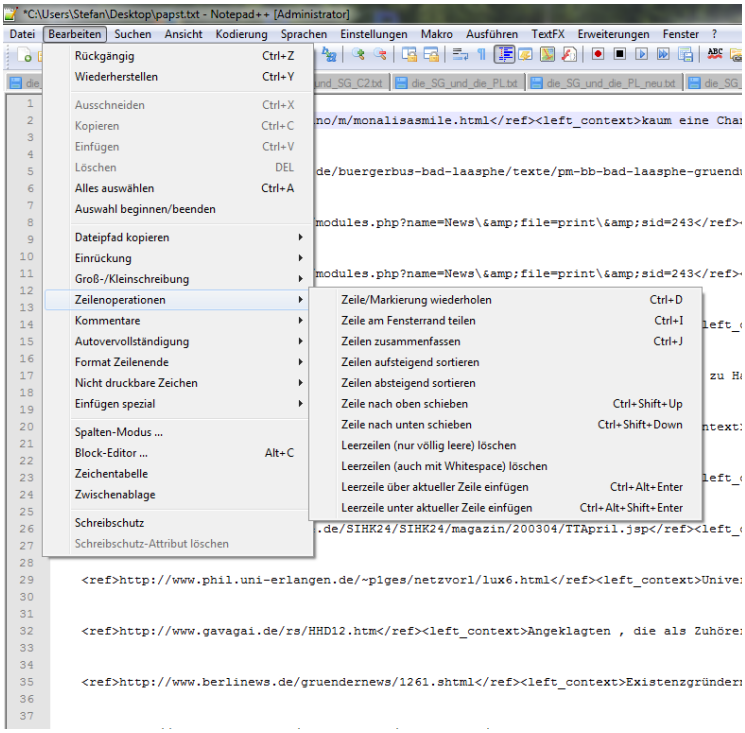
<ref> ... </ref>	Die Quelle.
<left_context> ... </left_context>	Kontext links vom Suchwort.
<kwic> ... </kwic>	Das Keyword/Suchwort selbst (z.B. *Literaturpapst*).
<right_context> ... </right_context>	Der Kontext, der auf das Keyword folgt.

Im neuen Textdokument, in das wir die Konkordanz ohne Header eingefügt haben, nutzen wir diese absolut regelmäßige Struktur, um eine Tabelle zu bekommen, in der die einzelnen Spalten durch **Tabs** separiert sind. (Ein Tab ist das, was Sie erhalten, wenn Sie ↹ auf der Tastatur drücken.)

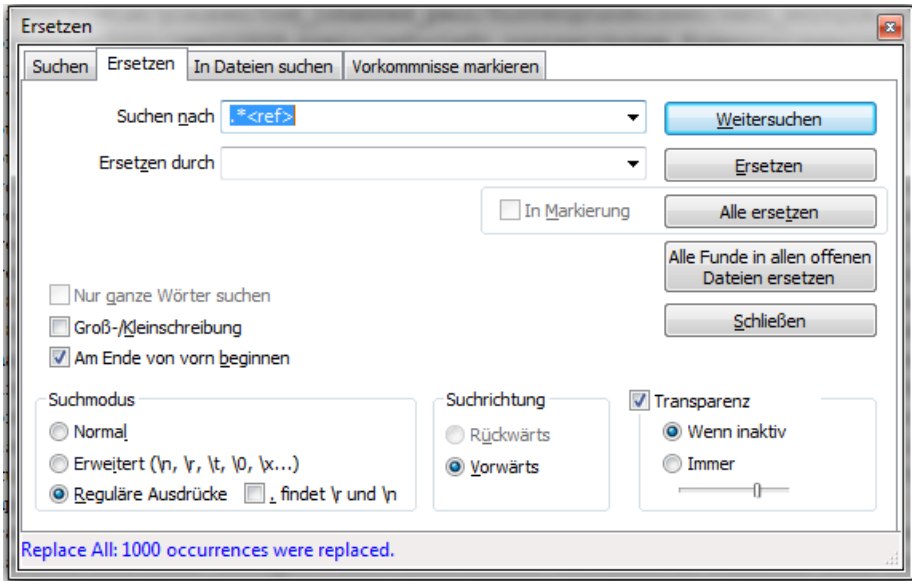
Zuerst wollen wir die <line>-Markierungen loswerden. Über Suche > Ersetzen ersetzen wir alle `<line>` und `</line>` im Dokument durch nichts.



Für diesen Vorgang benutzen wir den normalen Suchmodus. Nun haben wir natürlich eine ganze Menge leerer Zeilen. In aktuellen Notepad-Versionen können wir diese jedoch ganz einfach löschen, und zwar mit **Bearbeiten > Zeilenoperationen > Leerzeilen (auch mit whitespace) löschen.**



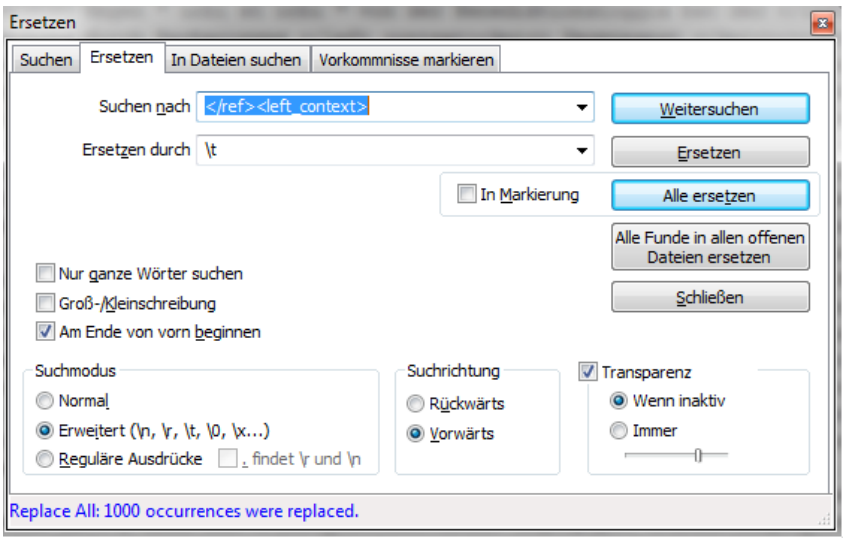
Sie sehen, dass nun jede Zeile mit einigen Leerzeichen und <ref> beginnt. Um beide loszuwerden, arbeiten wir wieder mit **regulären Ausdrücken**, die wir oben schon in CQL am Beispiel der Wiederholungsoperatoren (wie * und +) kennengelernt haben.



Um *regular expressions* nutzen zu können, müssen wir den entsprechenden Suchmodus wählen. Dann ersetzen wir .*<ref> - also *<ref>* mit einer beliebigen Anzahl von vorangehenden Zeichen (hier: Leerzeichen) - durch nichts. Et voilà, nun beginnt jede Zeile unmittelbar mit der Quelle.

Nun wollen wir die einzelnen Spalten durch **Tabs** abtrennen. Das ist sehr einfach: Wir wählen den Suchmodus „erweitert (\n, \r etc.)“. Der Ausdruck **\t** steht in diesem erweiterten Modus für einen Tab. Deshalb können wir die Abfolge von schließenden und öffnenden Tags einfach durch \t ersetzen. Zum Beispiel schließt die Angabe der Quelle ja, wie wir gesehen haben,

immer mit einem </ref>-Tag, dem unmittelbar ein <left_context>-Tag folgt. Also können wir </ref><left_context> einfach durch \t ersetzen:



Das gleiche wiederholen wir für </left_context><kwic> und </kwic><right_context>.
</right_context> wiederum können wir wieder durch nichts ersetzen.

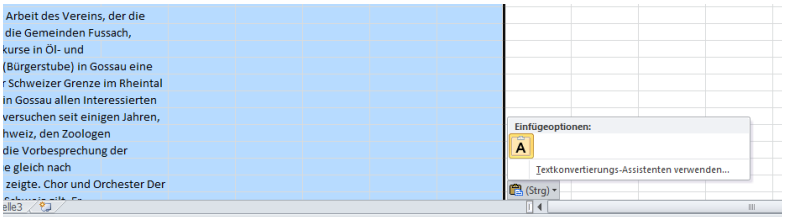
Zu guter Letzt entfernen wir noch manuell die letzten beiden Zeilen </lines> und </concordance> und haben eine wunderbare Tabelle, die wir direkt in Excel copy&pasten können.

5. In Excel: Struktur überprüfen

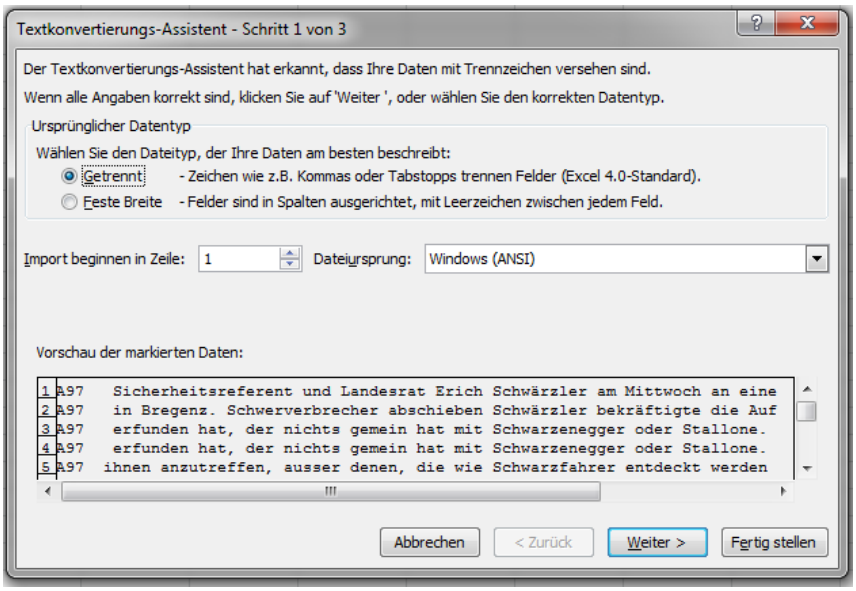
Apropos copy&pasten: Da das weitere Vorgehen mit dem bei anderen Korpora praktisch identisch ist, copy&paste ich hier meine Anleitung für COSMAS. Seien Sie daher nicht verwirrt, dass wir das *-papst*-Beispiel kurz verlassen und in den nächsten Bildern Komposita mit *Schwarz-* wie z.B. *Schwarzarbeiter* zu sehen sind.

Wenn Sie Ihre Tabelle in Excel ge-copy&pasted haben, überprüfen Sie zunächst, ob die Struktur stimmt. Sie sollte vier Spalten haben (Quelle, linker Kontext, Keyword, rechter Kontext) und genauso viele Zeilen wie das Notepad-Dokument. Ist das nicht der Fall, kann es sein, dass Excel Anführungszeichen in den Belegtexten als **Textqualifizierer** behandelt. Ist das der Fall, lesen Sie hier weiter, ansonsten springen Sie zum nächsten Abschnitt. (Soweit ich sehe, betrifft dieses Problem die WaCkY-Daten jedoch nicht, da die Satzzeichen hier jeweils durch Leerzeichen vom vorangehenden oder nachfolgenden Wort getrennt sind.)

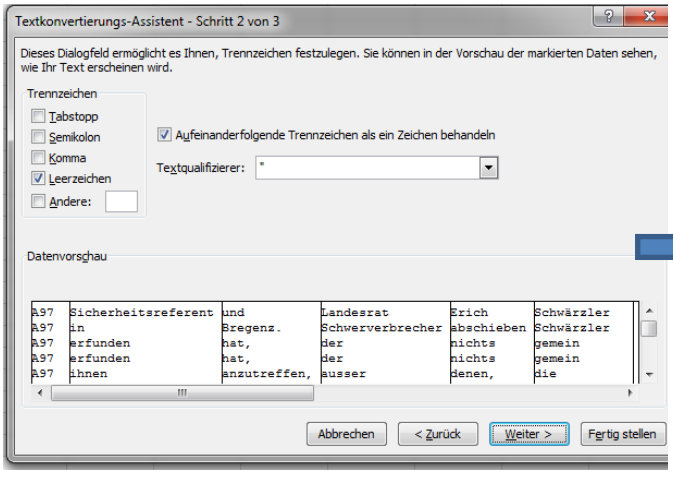
Anführungszeichen behandelt Excel oft defaultmäßig als **Textqualifizierer**, also als Marker, die Excel mitteilen, dass es die entsprechenden Daten als Text behandeln soll. So kommt es dazu, dass teilweise die Daten zwischen zwei Anführungszeichen als ein zusammenhängender Text behandelt werden, was zu Chaos in den Daten führt. Um dies zu verhindern, klicken wir gleich nach dem Einfügen auf das kleine Kästchen neben der Markierung, in dem „{Strg}“ steht (dieses verschwindet wieder, sobald wir die Markierung aufheben, deshalb gleich draufklicken!).

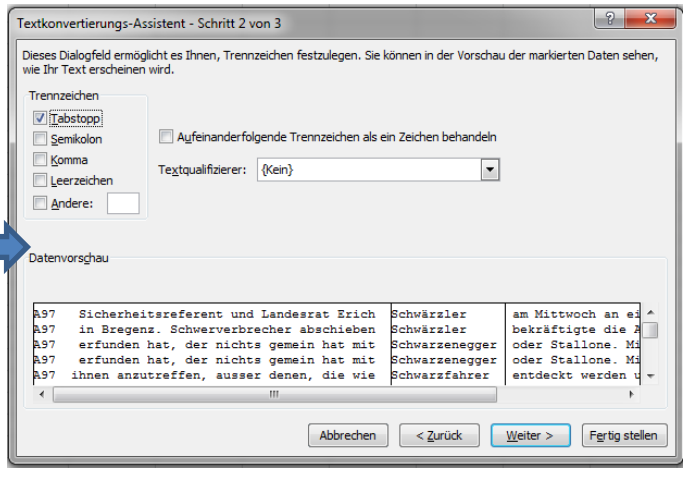


Hier wählen wir die Option „Textkonvertierungs-Assistenten verwenden“. Nun können wir Excel in drei Schritten mitteilen, wie es die Daten interpretieren soll:

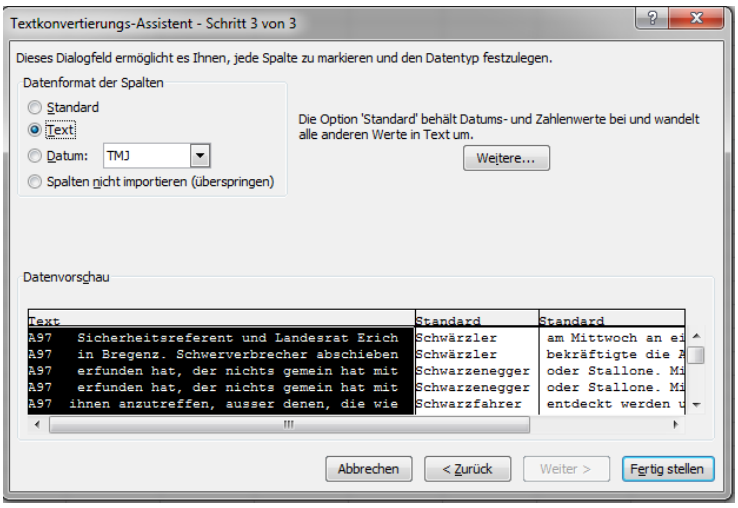


Im ersten Schritt ist die von uns gewollte Option „Getrennt“ bereits angewählt, wir können also einfach auf „Weiter“ klicken.





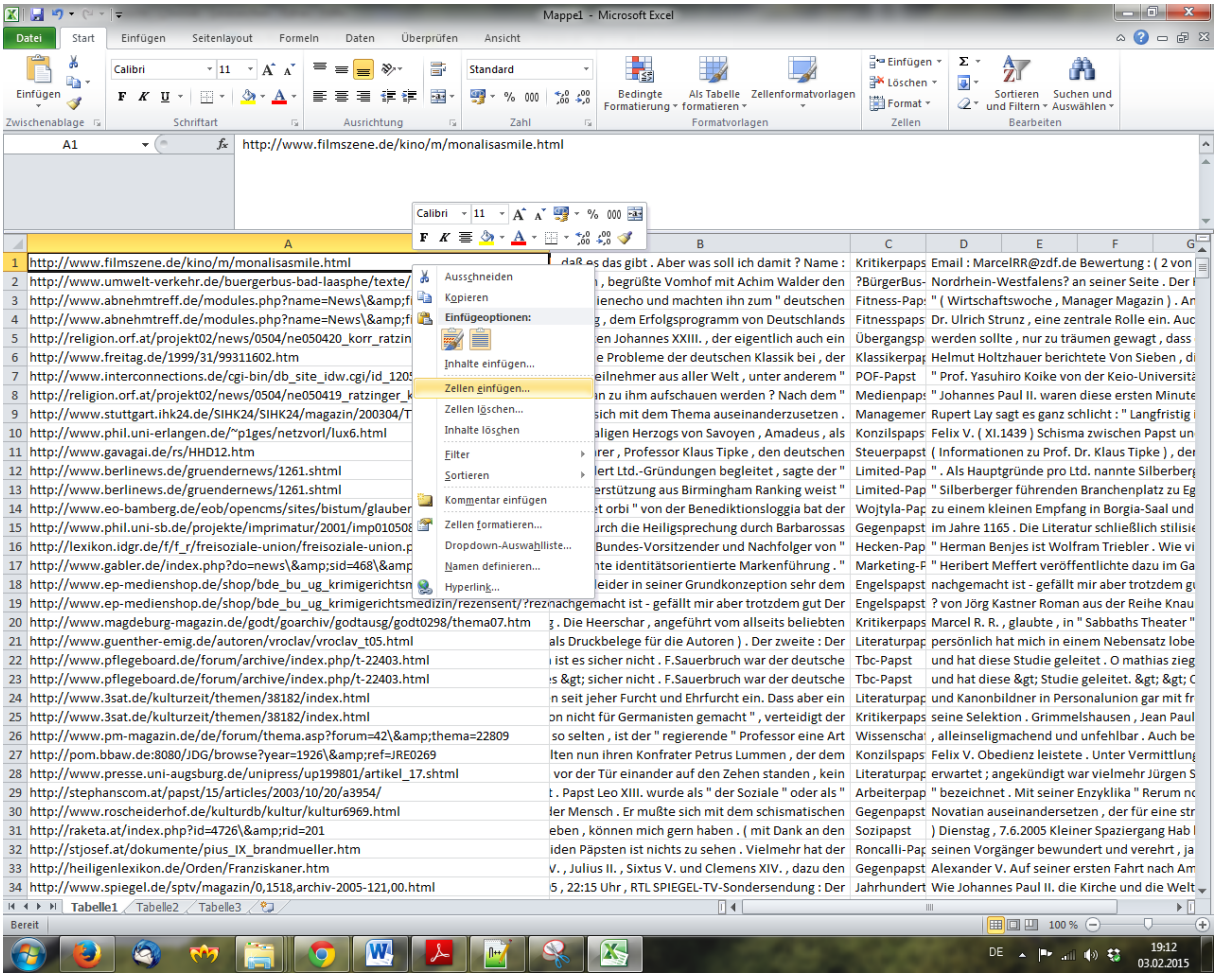
Im zweiten Schritt wählen wir **Tabstopp** als Trennzeichen und - ganz wichtig - deaktivieren die Erkennung von Anführungszeichen als Textqualifizierer, indem wir als **Textqualifizierer** *{kein}* auswählen.



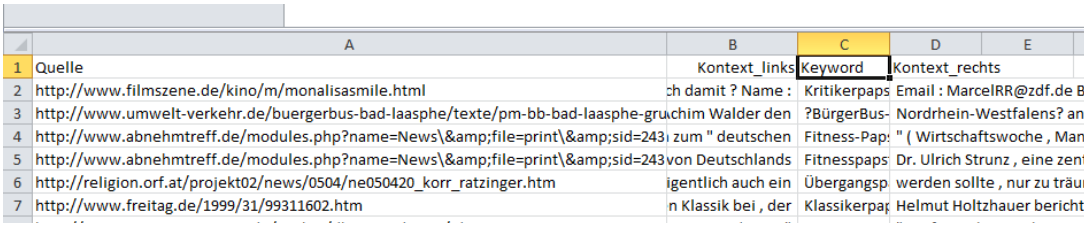
Im dritten Schritt schließlich können wir Excel noch mitteilen, dass es sich bei unseren Daten um **Text** handelt (nicht etwa um Zahlen oder um Kalenderdaten). Abschließend klicken wir auf **Fertig stellen.**

6. Sortieren der Excel-Tabelle

Um die Daten im Excel-Dokument nach Keyword sortieren zu können, fügen wir zunächst eine Überschriftenzeile ein: Rechtsklick in der ersten Zeile > Zellen einfügen > **Ganze Zeile.**

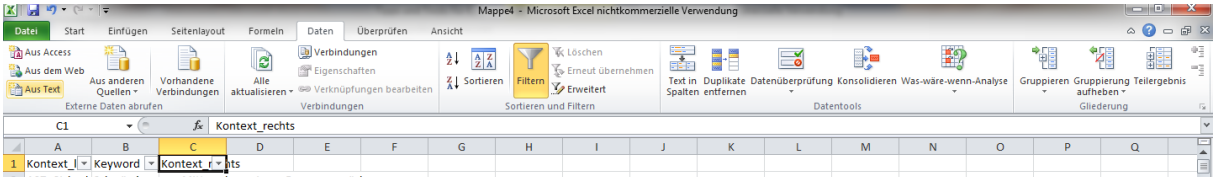


In die neu entstandene Zeile tragen wir nun die Überschriften für die einzelnen Spalten ein, z.B. Kontext_links, Keyword, Kontext_rechts.

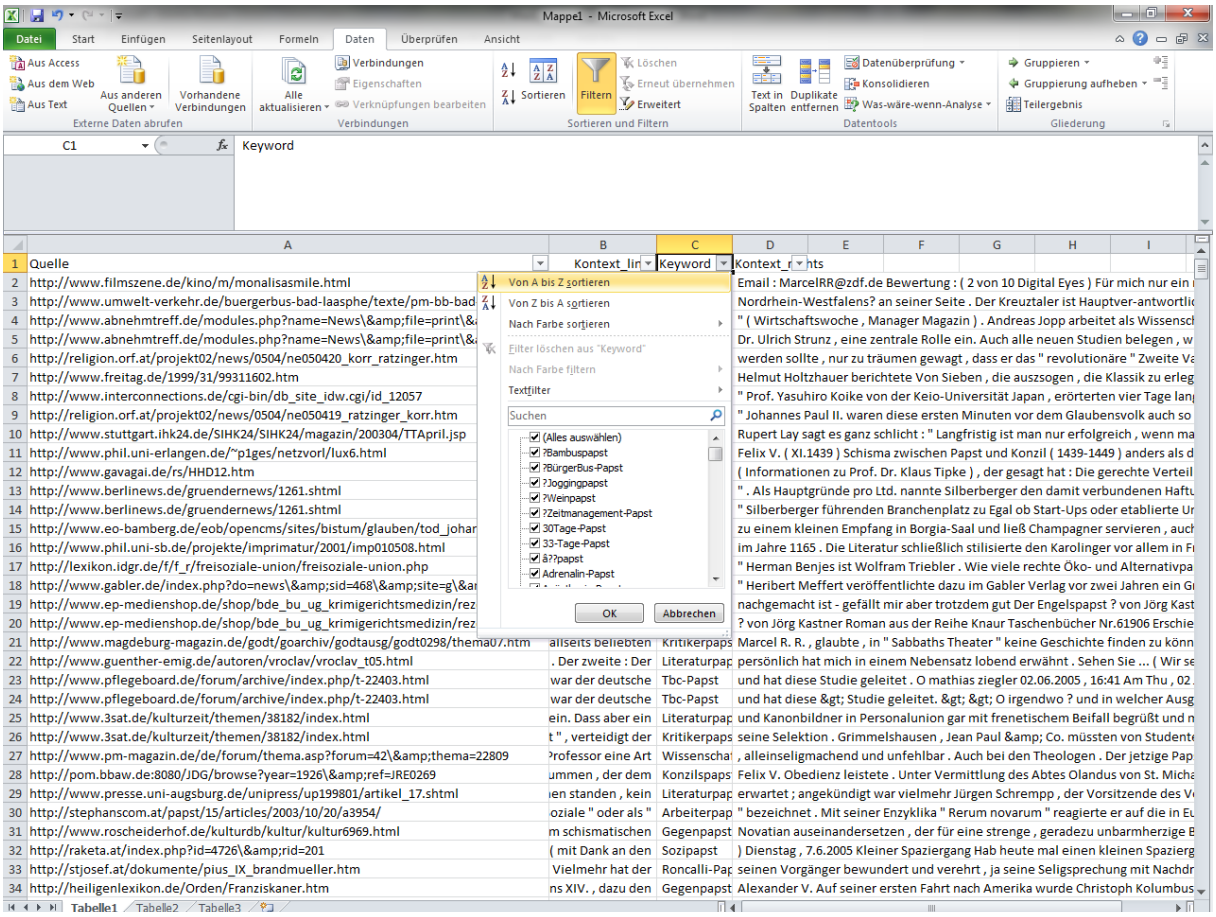


	A	B	C	D	E
1	Quelle	Kontext_links	Keyword	Kontext_rechts	
2	http://www.filmszene.de/kino/m/monalisasmile.html	ch damit ? Name :	Kritikerpaps	Email : MarcelRR@zdf.de B	
3	http://www.umwelt-verkehr.de/buergerbus-bad-laasphe/texte/pm-bb-bad-laasphe-gru	chim Walder den	?BürgerBus-	Nordrhein-Westfalens? an	
4	http://www.abnehmtreff.de/modules.php?name=News\&file=print\&sid=243	zum " deutschen	Fitness-Pap	" (Wirtschaftswoche , Man	
5	http://www.abnehmtreff.de/modules.php?name=News\&file=print\&sid=243	von Deutschlands	Fitnesspaps	Dr. Ulrich Strunz , eine zen	
6	http://religion.orf.at/projekt02/news/0504/ne050420_korr_ratzinger.htm	igentlich auch ein	Übergangsp	werden sollte , nur zu träu	
7	http://www.freitag.de/1999/31/99311602.htm	n Klassik bei , der	Klassikerpap	Helmut Holtzhauer bericht	

Wir lassen die erste Zeile markiert, gehen ganz oben auf den Reiter DATEN und dort auf **Filtern**.

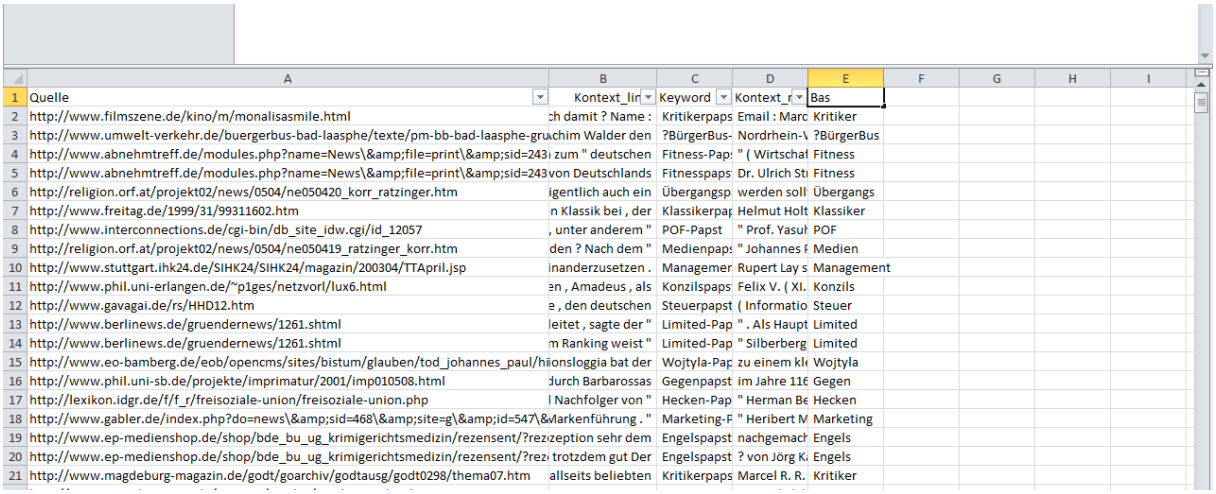


Nun sehen Sie kleine Pfeilchen neben den Überschriften. Mit einem Klick auf das Pfeilchen neben **Keyword** können wir die Schlüsselwörter „Von A bis Z sortieren“:



7. Und nun?

Nun können Sie die Daten ggf. noch nach Kriterien, die Sie interessant finden, annotieren. Auch können Sie sich die Erstglieder quantitativ näher anschauen, z.B. indem Sie die Keyword-Spalte wieder in Notepad copy&pasten, dort „papst“ durch „\tpapst“ ersetzen (Suchmodus: erweitert; Häkchen bei „Groß- und Kleinschreibung beachten“ muss entfernt sein), dann „\t-“ durch nichts ersetzen, um die Bindestriche loszuwerden und schließlich im Suchmodus *Reguläre Ausdrücke* „papst.*“ durch nichts ersetzen. Zuletzt können wir noch, ebenfalls im Suchmodus *Reguläre Ausdrücke,* „[[:punct:]]“ durch nichts ersetzen, um alle Interpunktionszeichen (z.B. in *?BürgerBus-Papst*) zu entfernen. Die verbleibenden Erstglieder können Sie nun in eine neue Spalte Ihres Excel-Dokuments copy&pasten.

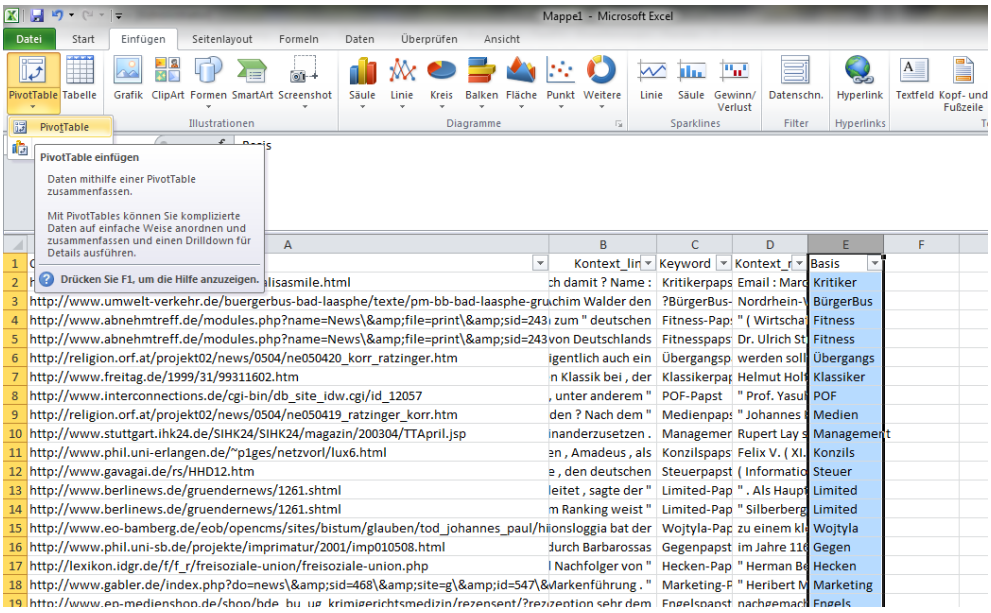


	A	B	C	D	E	F	G	H	I
1	Quelle	Kontext_lir	Keyword	Kontext_r	Bas				
2	http://www.filmszene.de/kino/m/monalisasmile.html	ch damit ? Name :	Kritikerpaps	Email : Marc	Kritiker				
3	http://www.umwelt-verkehr.de/buergerbus-bad-laasphe/texte/pm-bb-bad-laasphe-gru	chim Walder den	?BürgerBus-	Nordrhein-\	?BürgerBus				
4	http://www.abnehmtreff.de/modules.php?name=News\&file=print\&sid=243	zum " deutschen	Fitness-Paps	" (Wirtschaf	Fitness				
5	http://www.abnehmtreff.de/modules.php?name=News\&file=print\&sid=243	von Deutschlands	Fitnesspaps	Dr. Ulrich Sti	Fitness				
6	http://religion.orf.at/projekt02/news/0504/ne050420_korr_ratzinger.htm	igentlich auch ein	Übergangsp	werden soll	Übergangs				
7	http://www.freitag.de/1999/31/99311602.htm	n Klassik bei , der	Klassikerpaj	Helmut Holt	Klassiker				
8	http://www.interconnections.de/cgi-bin/db_site_idw.cgi/id_12057	, unter anderem "	POF-Papst	" Prof. Yasul	POF				
9	http://religion.orf.at/projekt02/news/0504/ne050419_ratzinger_korr.htm	den ? Nach dem "	Medienpaps	" Johannes I	Medien				
10	http://www.stuttgart.ihk24.de/SIHK24/SIHK24/magazin/200304/TTApril.jsp	inanderzusetzen .	Managemer	Rupert Lay s	Management				
11	http://www.phil.uni-erlangen.de/~p1ges/netzvorl/lux6.html	en , Amadeus , als	Konzilspaps	Felix V. (XI.	Konzils				
12	http://www.gavagai.de/rs/HHD12.htm	e , den deutschen	Steuerpapst	(Informatio	Steuer				
13	http://www.berlinews.de/gruendernews/1261.shtml	leitet , sagte der "	Limited-Pap	" . Als Haupt	Limited				
14	http://www.berlinews.de/gruendernews/1261.shtml	m Ranking weist "	Limited-Pap	" Silberberg	Limited				
15	http://www.eo-bamberg.de/eob/opencms/sites/bistum/glauben/tod_johannes_paul/hi	onsloggia bat der	Wojtyla-Pap	zu einem kl	Wojtyla				
16	http://www.phil.uni-sb.de/projekte/imprimatur/2001/imp010508.html	durch Barbarossas	Gegenpapst	im Jahre 116	Gegen				
17	http://lexikon.idgr.de/f/f_r/freisoziale-union/freisoziale-union.php	l Nachfolger von "	Hecken-Pap	" Herman Be	Hecken				
18	http://www.gabler.de/index.php?do=news\&sid=468\&site=g\&id=547\&	Markenführung . "	Marketing-F	" Heribert M	Marketing				
19	http://www.ep-medienshop.de/shop/bde_bu_ug_krimigerichtsmedizin/rezensent/?rez	zeption sehr dem	Engelspapst	nachgemach	Engels				
20	http://www.ep-medienshop.de/shop/bde_bu_ug_krimigerichtsmedizin/rezensent/?rez	trotzdem gut Der	Engelspapst	? von Jörg Ki	Engels				
21	http://www.magdeburg-magazin.de/godt/goarchiv/godtausg/godt0298/thema07.htm	allseits beliebten	Kritikerpaps	Marcel R. R.	Kritiker				

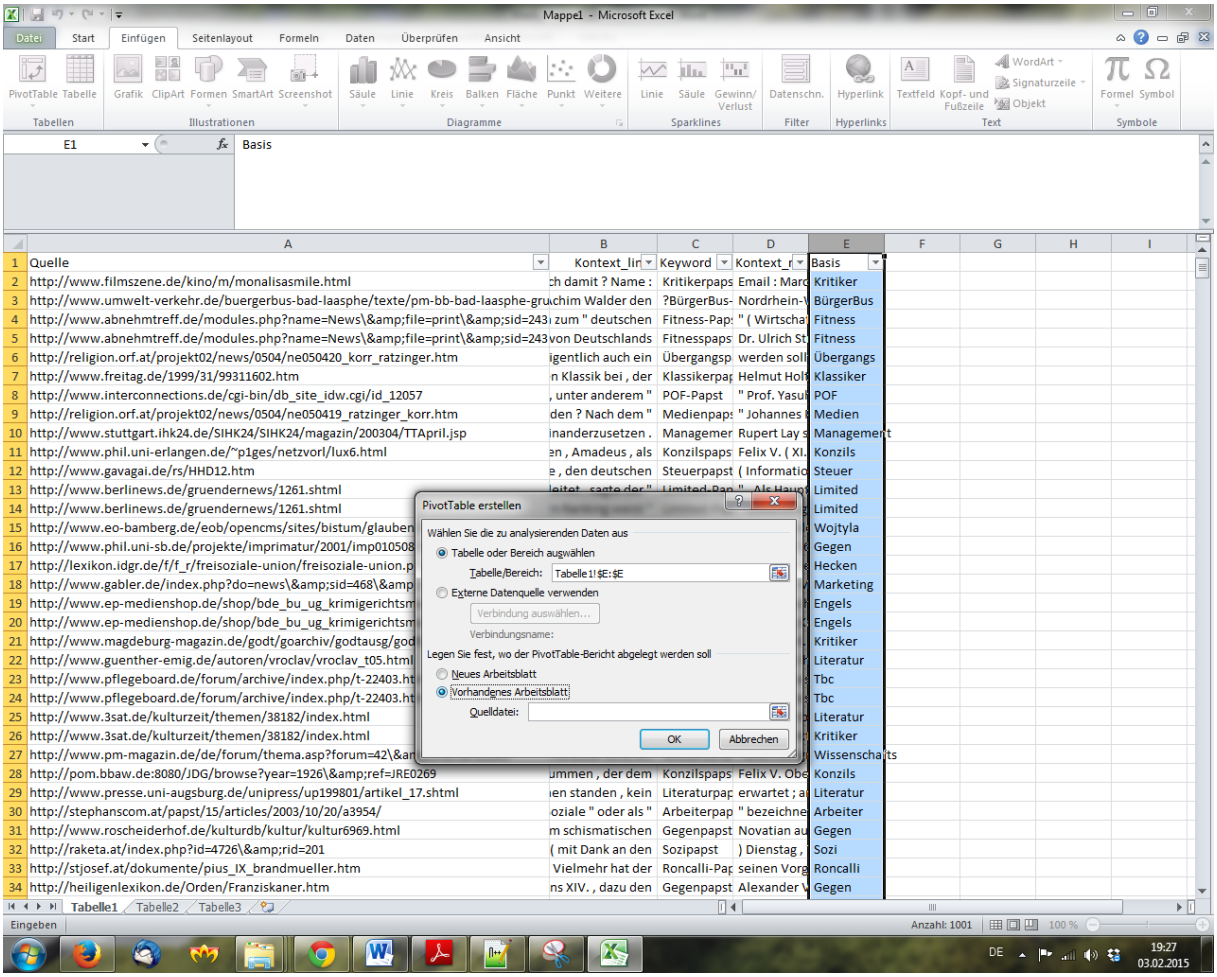
In der neuen Spalte ersetzen wir die Überschrift „Keyword“ durch „Basis“ (sonst hätten wir ja zwei Spalten mit der gleichen Überschrift).

Diese neue Spalte ist zunächst nicht gefiltert, aber indem wir den Filter oben deaktivieren und dann wieder aktivieren, stellen wir sicher, dass die gesamte Zeile gefiltert wird und wir auch die Lesart, wenn wir die entsprechende Spalte annotiert haben, alphabetisch sortiert und eben auch gefiltert werden kann.

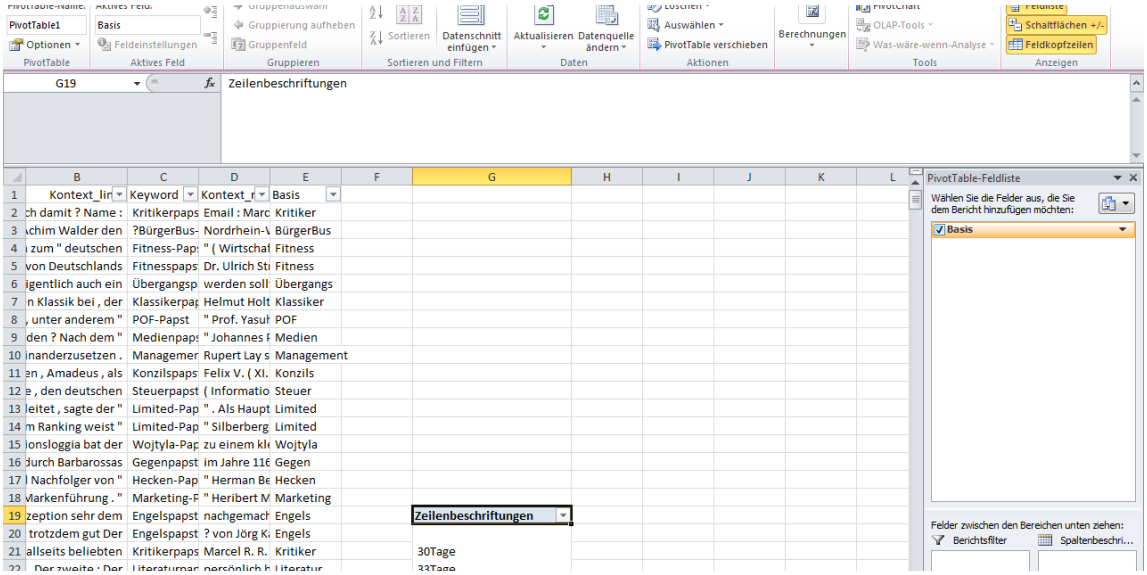
Auch können wir nun eine **Pivot-Tabelle** erstellen. Dafür gehen wir auf den Reiter Einfügen>PivotTable>PivotTable, nachdem wir die gesamte neue Spalte markiert haben.



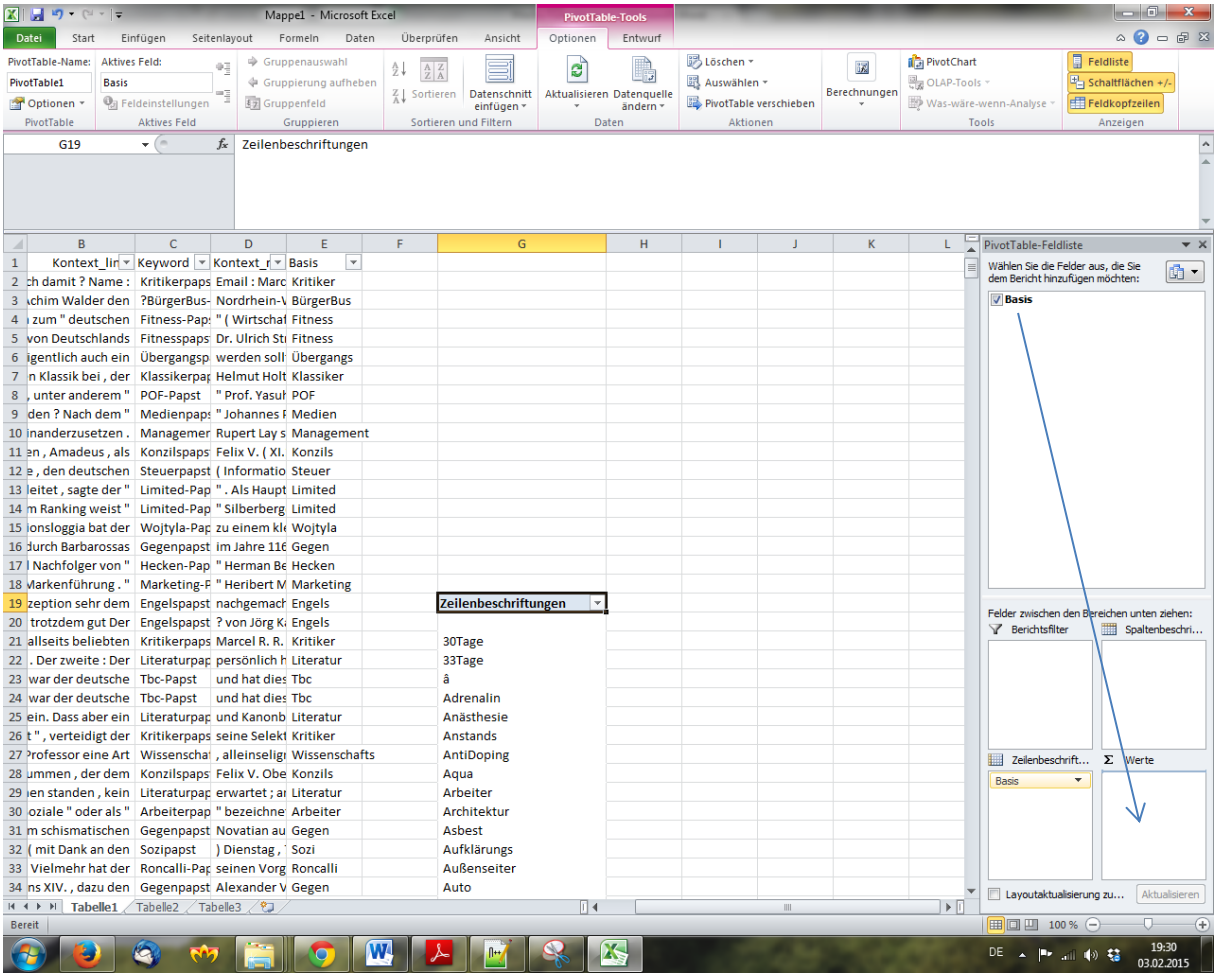
Wir können die Tabelle in einem neuen oder im vorhandenen Arbeitsblatt einfügen - hier wählen wir einfach das vorhandene.



Bevor wir auf OK klicken, müssen wir auswählen, wo die Tabelle eingefügt werden soll, indem wir auf irgendeine der leeren Zellen rechts klicken.
Es erscheint die „PivotTable-Feldliste“, in der wir die Tabelle aktivieren, indem wir bei „Basis“ ein Häkchen setzen:



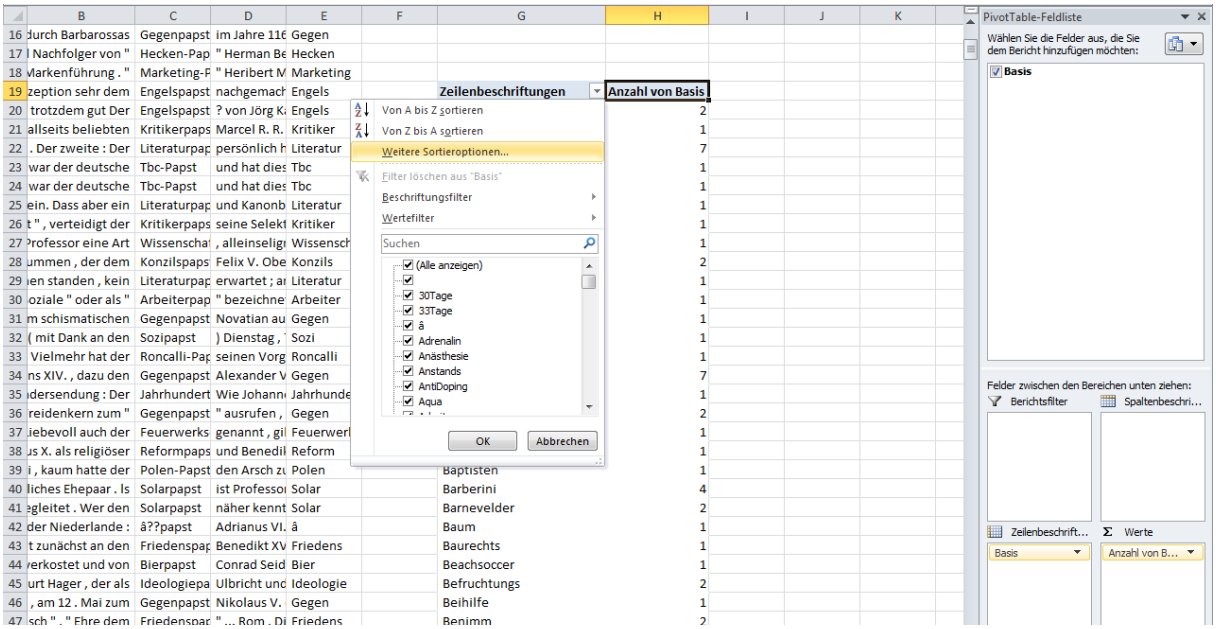
Dann klicken wir auf „Basis“, halten die Maustaste gedrückt und ziehen es nach unten rechts in das Feld „Werte“.



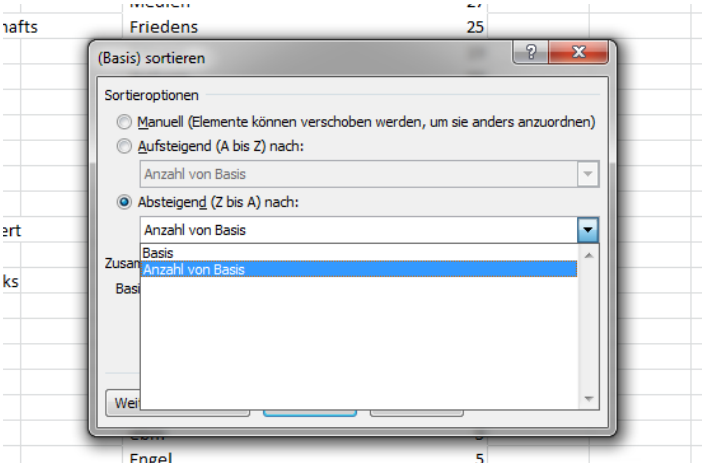
Nun sehen wir, dass Excel die einzelnen Types für uns gezählt hat:

Zeilenbeschriftungen	Anzahl von Basis
	2
30Tage	1
33Tage	7
â	1
Adrenalin	1
Anästhesie	1
Anstands	1
AntiDoping	1
Aqua	2
Arbeiter	1
Architektur	1
Asbest	1
Aufklärungs	1

Sie sind jedoch noch in alphabetischer Reihenfolge. Um sie nach Häufigkeit zu sortieren, klicken wir auf das Pfeilchen neben „Zeilenbeschriftungen“ und wählen „Weitere Sortieroptionen“.



In dem sich öffnenden Dialogfeld gehen wir auf „Absteigend: Z bis A nach“ und wählen aus der Liste darunter „Anzahl von Basis“ aus.



Nun erhalten wir eine Tabelle, der wir entnehmen können, dass der historische *Gegenpapst* die häufigste Bildung ist, aber der *Literaturpapst* schon auf Rang 2 folgt. Innovative Bildungen wie *Bierpapst* sind überraschend tokenfrequent - hier wäre noch zu prüfen, ob sich evtl. Dubletten eingeschlichen haben (das geschieht bei WaCkY oft, etwa wenn in einem Forenthread ein Beitrag zitiert wird und dann der Originalbeitrag als auch das Zitat, oder sogar mehrfache Zitate, Eingang ins Korpus finden).

Zeilenbeschriftungen	Anzahl von Basis
Gegen	140
Literatur	98
Engels	37
Übergangs	31
Bier	29
Kritiker	28
Medien	27
Friedens	25
Fitness	19
Reform	15
Kaffeeautomaten	12
Konzils	10
Marketing	10
Reise	9
Renaissance	8
Solar	7
Auto	7
33Tage	7
Wander	6
Jazz	6
Wojtyla	6
Messe	6
Ernährungs	5
ebm	5
Engel	5
Kultur	5
Wein	5
Management	5
Ideologie	4
Jugend	4

Soweit das Tutorial zu WaCkY, bei dessen Erstellung ich selbst sehr viel Neues über dieses Tool gelernt habe. Für Verbesserungsvorschläge zu diesem Dokument bin ich natürlich jederzeit dankbar!