Open Boek: technical report and manual, version 3.0

Hans Paijmans, Sander Wubben, Alex Brandsen

Abstract

This report provides a description of the Open Boek intelligent retrieval system version 3, and of its care and feeding. It combines the user manual and the administration guide. Finally, it provides detailed descriptions of the scripts and fileformats.

Contents

1 Introduction

Open Boek is the 'use case' of the two CATCH projects RICH and MITCH. It aims ultimately at the extraction and combination of textual and visual data from written documents so that databases of images and corresponding data can be created from reports in natural language. As a first stage, we implemented a system that can recognize the semantics of numeric data for, e.g. chronological search and retrieval[4, 5, 2].

This report is a description of this first stage of Open Boek. It provides information for the end user, for the administrator and for hackers who want to improve or enlarge the system. Our programs and scripts are published under the GNU license, but please note that SMART, TiMBL and perhaps other programs are published under different conditions, although the source of everything that is directly related to Open Boek is available.

The end user will want to skip the technical details and only read section 2. This is why we put this section right below. The administrator should read the two following sections about installation and indexing. If you want to change the system, or want to change how it works, read everything.

1.1 Versions

Version 3 is a complete overhaul of version 2, in which the language dependent modules are overhauled and prepared for english and german. In version 2, MySQL is added to the packages that should be installed. The retrieval functions are separated from the administration functions, so that the data and indexes can be burned on a CD or DVD.

Version 1 differs from version 0 for the most part in that the individual html-files are discarded in favour of stand-off organisation, where tokens and tags are stored in different files, and are only combined at display time. This should improve the speed of indexing. Also, the directory system is overhauled, so that a single installation of Open Boek can access several databases. Finally, we added an annotation tool (see section 7) so that the user can create or tune the MBL data for his own databases.

DISCLAIMER No warranties are given as to the performance of Open Boek and its useability in certain areas. This manual naturally lags behind the development. Differences between the description of the system on these pages and the real thing may and will occur. Your Mileage May Vary.

1.2 Copyrights

Open Boek itself is as all CATCH projects published under the GPL3, as are most of the Linux and Unix applications that it uses. In other words, it is Open Source and free. Some of the programs that we use, such as

Timble or SMART, have different conditions: it is your responsibility to learn those conditions and complay with them.

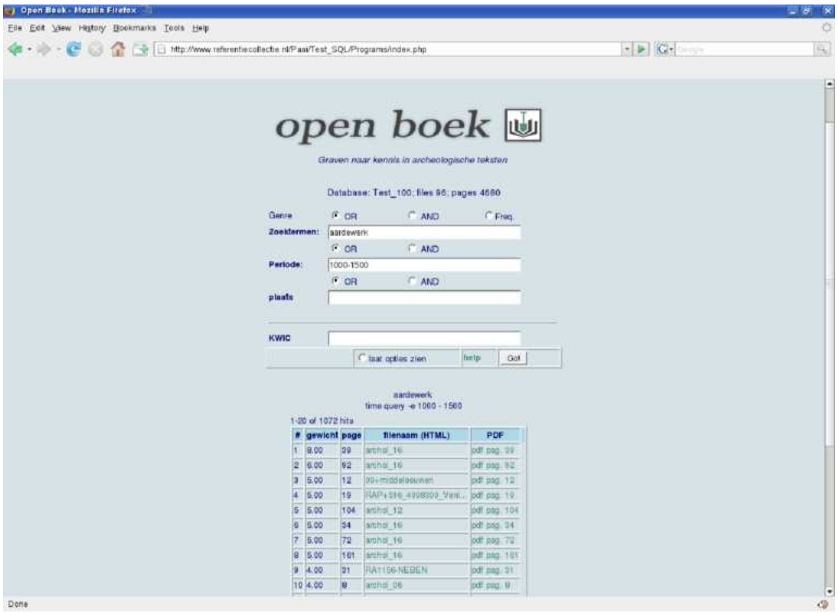
2 User manual

Here we describe the user interface to Open Boek and the details of retrieval within the system in some detail. First, we cover the keyword retrieval, the chronological retrieval and the KWIC retrieval. Then we will tell you how to arrange the retrieved pages to best effect.

Apart from this graphical interface, you can have access to the retrieval functionality by adding parameters to the URL, so you can develop your own interfaces. This will be described in more detail in section 3.

2.1 Selecting a database and simple retrieval

When Open Boek is opened by entering the URL in a browser, the system first lists the available databases and their state of indexing. You select one and the browser will display the search interface (see figure 1). If the database is marked as 'indexed on pages' only, you can only list and display the files (with the special query 'filelist') but not search for keywords, place names or chronology.



Open Boek - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://www.referentiecollectie.nl/Paai/Test_SQL/Programs/index.php
open boek
Graven naar kennis in archeologische teksten
Done

Database: Test_100; files 96; pages 4680

Genre ● OR ○ AND ○ Freq.

Zoektermen: aardewerk

● OR ○ AND

Periode: 1000-1500

● OR ○ AND

plaats

KWIC

○ laat opties zien help Go!

aardewerk

time query -e 1000 - 1500

1-20 of 1072 hits

#	gewicht	page	filenaam (HTML)	PDF
1	8.00	39	archol_16	pdf pag. 39
2	6.00	92	archol_16	pdf pag. 92
3	5.00	12	99+middeleeuwen	pdf pag. 12
4	5.00	19	RAP+316_4998300_Verl...	pdf pag. 19
5	5.00	104	archol_12	pdf pag. 104
6	5.00	34	archol_16	pdf pag. 34
7	5.00	72	archol_16	pdf pag. 72
8	5.00	161	archol_16	pdf pag. 161
9	4.00	31	RA1156-NEBEN	pdf pag. 31
10	4.00	8	archol_06	pdf pag. 8

Figure 1: User interface

Retrieval in Open Boek is very simple: just type the keywords in the spaces provided and press 'submit'. After a few moments you will be presented with a list of links that point to pages or documents that may be relevant to your query. Click on the link, and you will see the text of the page. Words or phrases in that text that caused the document to be flagged as relevant, are in red. It is possible to search for keywords, for timespans or for geographic locations, and for combinations of any or all three *semantic concepts*.

For every concept there is a separate inputfield (see fig. 1). There is also a simple syntax to enter timespans and locations directly in the first inputfield: see the paragraphs , 2.2, 2.2, 2.3 and 2.4, 2.7.

Finally there is a reserved word: *filelist*. If you enter the word *filelist* as keyword, you will get a list with *all* documents in the database.

2.2 Keyword search

Keywords are just typed in the first inputfield, preferably without operators such as AND or OR. Never forget that the more keywords you enter, the higher the chances to find relevant information. Overload is largely avoided by ordering the results of your query on estimated relevance. See below for an explanation of the three ways they can be combined, weighted and sorted.

Wildcards

A recurring problem in keyword retrieval is that of *homonyms*, words that are spelled similarly, but mean very different things, like 'bow', which may be either the front part of a ship, an instrument to shoot arrows or the act of bending before a king[1]. That is why we encourage you to describe your information need in several words: 'bow waves sea' will bring you to pages about the nautical meaning, whereas 'bow arrow' will get you to Robin Hood. That may look obvious, but research has shown that the average query on e.g. Google is shorter than two words (1.7 to be precise), and are for the most part four letters long.

Dutch, english and most other European languages use suffixes for plurals and other variations. To avoid typing in all variations, you can just type the beginning of a word, followed by an asterisk, and all variations will be included in the search. So 'bow*' as query will get you 'bow', 'bows', 'bowing', 'bowman' but also 'bowl' and 'bowel'. In the same vein, the point ('.') is used for a single character: 'd.gger' will expand to 'dagger' and 'digger'. Of course you can combine both wildcards.

Relevancy

As we said under 'keyword retrieval', the list with links is sorted according to relevance, but what exactly is relevance? The answer is that we don't know. Or rather: relevance varies so wildly with the needs of the user, that it is very difficult to capture in a single trick or formula. Open Boek

[1] Please understand that OB is trained on the dutch language, but in this manual we have translated all dutch examples to their english equivalents

offers three different ways to rank the retrieved documents on estimated relevance and we urge you strongly to experiment with them.

AND We already mentioned the problem with homonyms. But apart from that, if you are interested about information about Tom, Dick and Harry, should the system assume that the pages where all three names Tom, Dick and Harry occur together will be more interesting to you than pages with only Tom and Harry? Yes, that seems obvious. Open Book will indeed assume that this is so and offer **'AND'** as the default option. This means that Open Boek will only find documents containing Tom **AND** Dick **AND** Harry. In expert search, you use the keyword 'AND'.

OR The 'AND' option will only give you your exact query, and will rank those accordingly. The **'OR'** option will produce a result in wich documents containing Tom, Dick and Harry will get a high ranking (three in this example), a document containing only two of those names will get a medium ranking (two), and documents containing only 1 of the names will get a low ranking (one). In expert search you use 'OR'.

Freq. The option described above does not take in account the frequency of the individual words. But why would we want to do that? Because we may assume that the more often the word 'Harry' occurs on a page, the more important the concept is (for that particular page). If you select **'Freq.'** as option, Open Boek will take the frequency of the keywords into account when it ranks the pages on relevance. Interestingly, it now is possible that pages with many Toms and Dicks, but without mention of Harry, wil rank as more relevant than a page where all three, Tom, Dick and Harry, are mentioned just once.

All these esoteric tricks and twists cause 'interesting words' to rank higher than relatively uninteresting words and as we said before, it is a good idea to experiment with these options.

2.3 Chronological search

Apart from searching by keywords, it is also possible to search on chronological dates. Indeed this is one of the reasons why you would use Open Boek. Searching on dates is as simple as entering the name of an era or period, or a range of years (in arabics) in the field provided. For example "1000-1500", "450BC-100", "10000BP-0" or "Neolithic" are all correct queries. Between the years of the timespan a minus sign is used (-). Note the suffix "BC" for years before Christ and the suffix "BP" for years before present (present being 1950). For the impatient, the ';' operator can be used to enter period and keywords in the first input field (table 2.7 lines 4, 5 and 8).

Open Boek 'knows' what time is and what years are, and will return all pages with dates that fall within the range you entered, regardless how they are written in the document. 'Twelfth century', '1100-1200' and '+XII AD' and its variations should all be recognized. Just as Open Boek 'knows' what time is, it also 'knows' wich dates are taken from references and bibliographies. By default, Open Boek will not recognise these dates as 'real' timespans, and will not include these in the search results.

By default a range in the document should fall entirely within the period you entered, that is: if you enter 1000 to 1500, it will *not* return pages with 'middle ages'. This is because the middle ages are defined as 500 - 1500, and to retrieve them, you should enter a start date equal to or less than 500 and an end date equal to or greater than 1500. However, the *late* middle ages are defined as between 1000 and 1500 and that will be retrieved, as will be every period or individual date between 1000 and 1500 inclusive (see below for how such names of eras are recognized).

The operator @ (the 'at'-sign) changes this behaviour (table 2.7 line 3) . If a timespan is preceeded by this sign, a document will be flagged as a hit if a period in the document starts or ends in the timespan indicated by the query. '@1000-1500' will return all timespans that begin or end in that period, so now the middle ages will be retrieved.

In table 2.7 line 5 we have shown that you can enter a named period in a timespan; in line 6 the use of BC is demonstrated (BP is allowed too, where present is 1950). You can inspect the list with named periods in *eras.rc* (table. 8). Modifications and extensions of this list should be left to the administrator (see section 3).

The recognition of chronological dates is a function of so-called artificial intelligence, and like human intelligence it will occasionally be wrong. In most of those cases where it errs, other numbers in the text are wrongly marked as years.

2.4 Geographical search

Names of cities, villages and other geographic entities obviously can be searched as keywords. However, if you want to make use of features as distance or area search, you need extra tools.

- Distance search. The location (e.g. Amersfoort) is considered a point, and you can search for other points within a circle with a given radius. To do this, enter the location in the keyword field and add the distance in kilometers between parentheses: "Amersfoort(17)" (see table 2.7 lines 7 and 8). There is also an opportunity to enter coordinates in stead of a geographical name.
- Area search (not yet implemented). The location is a polygon, and the search is for coordinates that lie within that polygon. Open questions are how the polygon is stored in the index, and how a point inside that polygon is defined.

- Disambiguation of geographical locations (not yet implemented).

Finally, Open Boek already recognizes spatial coordinates and is able to display the corresponding Google Maps. To do this, you just click on the link and Googlemaps will open in a new window. Of course, your administrator must have Googlemaps enabled on your site.

Important! Open Boek will try to ignore place names in literature references by default. This is because of the fact that the publishing information almost always contains a place name. Therefore you can not use this search feature if you expect to retrieve place names in booktitles, and you will have to enter such place names in the keyword field, which will show you *every* occurrence of the word.

Finally, spaces in place names should be replaced by underscores; use "Den_Bosch" in stead of "Den Bosch".

2.5 KWIC index

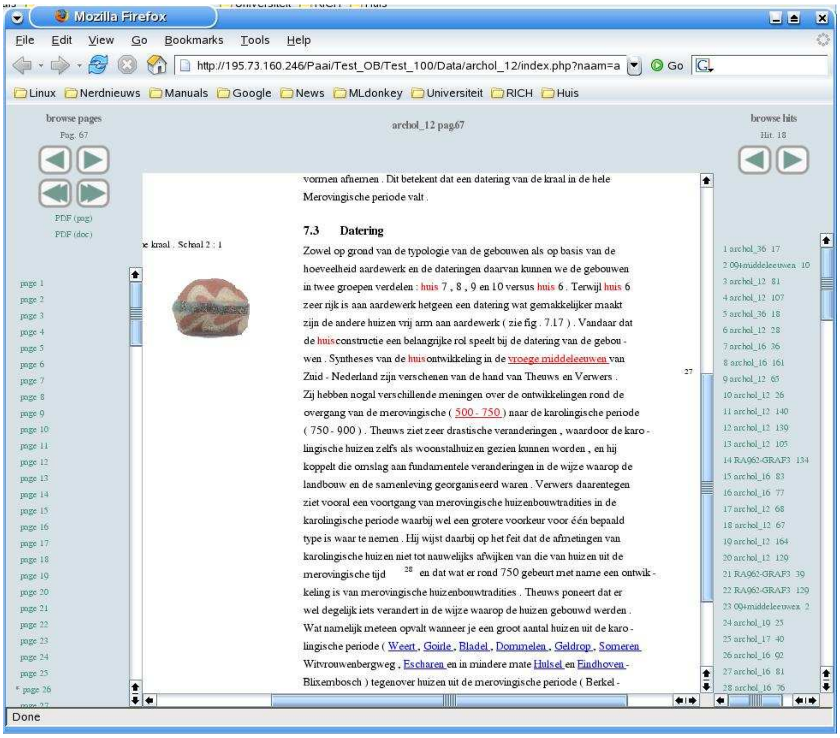
If you enter a string in this field, a KWIC index is generated. A KWIC (KeyWord In Context) Index shows the keyword in its context. The searching algorithm for the KWIC index function does not depend on the index of (single) keywords, but scans the full text of the documents. Therefore it is possible to define a query that includes spaces and other interpunction (but note that all interpunction should be separated by a space. If you want to search for a single word using the KWIC index, surround it with a space on either side. At the end of the table with KWIC index results, you will find a link to download the KWIC index for later reference.

The scanning of the full text may take some time on large document collections. After the first scan, the files reside in the cache, and subsequent scans during the same session go much faster.

2.6 Presentation of the results

In picture 1 you will see the results of the search in one table, consisting of 20 hits, each with it's corresponding attributes. From left to right; the number of the hit, the 'weight' of that page, the file name for the HTML version of the page and last, the link to the original PDF. The links will cause new windows to be opened with either the page that is referred to in pdf (column 'PDF') or with the page viewer, a tool for easy browsing of the search results (see fig. 2).

If the text has been extracted from a pdf-file, you can inspect the either the original page of that pdf-file or the complete file by clicking on the button **pdf (pag)** resp. **pdf (doc)** in the upper left frame of the window where the document is displayed. Here you also will find navigation buttons to browse through the complete document. Individual pages within the current document are accessed by clicking in the lefthand frame. At the righthand side you will see a similar frame. Here you can


Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://195.73.160.246/Paai/Test_OB/Test_100/Data/archol_12/index.php?naam=a
Go
Linux Nerdnieuws Manuals Google News MLdonkey Universiteit RICH Huis
browse pages
Pag. 67
PDF (pag)
PDF (doc)
archol_12 pag67
browse hits
Hit. 18
e kraal . Schaal 2 : 1
27
Done

page 1
page 2
page 3
page 4
page 5
page 6
page 7
page 8
page 9
page 10
page 11
page 12
page 13
page 14
page 15
page 16
page 17
page 18
page 19
page 20
page 21
page 22
page 23
page 24
page 25
* page 26

vormen afnemen . Dit betekent dat een datering van de kraal in de hele Merovingische periode valt .

7.3 Datering

Zowel op grond van de typologie van de gebouwen als op basis van de hoeveelheid aardewerk en de dateringen daarvan kunnen we de gebouwen in twee groepen verdelen : huis 7 , 8 , 9 en 10 versus huis 6 . Terwijl huis 6 zeer rijk is aan aardewerk hetgeen een datering wat gemakkelijker maakt zijn de andere huizen vrij arm aan aardewerk (zie fig . 7.17) . Vandaar dat de huisconstructie een belangrijke rol speelt bij de datering van de gebouwen . Syntheses van de huisontwikkeling in de vroege middeleeuwen van Zuid - Nederland zijn verschenen van de hand van Theuws en Verwers . Zij hebben nogal verschillende meningen over de ontwikkelingen rond de overgang van de merovingische (500 - 750) naar de karolingische periode (750 - 900) . Theuws ziet zeer drastische veranderingen , waardoor de karolingische huizen zelfs als woonstalhuizen gezien kunnen worden , en hij koppelt die omslag aan fundamentele veranderingen in de wijze waarop de landbouw en de samenleving georganiseerd waren . Verwers daarentegen ziet vooral een voortgang van merovingische huizenbouwtradities in de karolingische periode waarbij wel een grotere voorkeur voor één bepaald type is waar te nemen . Hij wijst daarbij op het feit dat de afmetingen van karolingische huizen niet tot nauwelijks afwijken van die van huizen uit de merovingische tijd [28] en dat wat er rond 750 gebeurt met name een ontwikkeling is van merovingische huizenbouwtradities . Theuws poneert dat er wel degelijk iets verandert in de wijze waarop de huizen gebouwd werden . Wat namelijk meteen opvalt wanneer je een groot aantal huizen uit de karolingische periode (Weert , Goirle , Bladel , Dommelen , Geldrop , Someren Witvrouwenbergweg , Escharen en in mindere mate Hulsel en Eindhoven - Blixembosch) tegenover huizen uit de merovingische periode (Berkel -

1 archol_36 17
2 09+middeleeuwen 10
3 archol_12 81
4 archol_12 107
5 archol_36 18
6 archol_12 28
7 archol_16 36
8 archol_16 161
9 archol_12 65
10 archol_12 26
11 archol_12 140
12 archol_12 139
13 archol_12 105
14 RA962-GRAF3 134
15 archol_16 83
16 archol_16 77
17 archol_12 68
18 archol_12 67
19 archol_12 164
20 archol_12 129
21 RA962-GRAF3 39
22 RA962-GRAF3 129
23 09+middeleeuwen 2
24 archol_19 25
25 archol_17 40
26 archol_16 92
27 archol_16 81
28 archol_16 76

Figure 2: Display window

navigate the list with 'hits', pages or documents that conform to your query. If the frame is too narrow to display the title and the page, please use the interface of your browser to adjust the width of the frame.

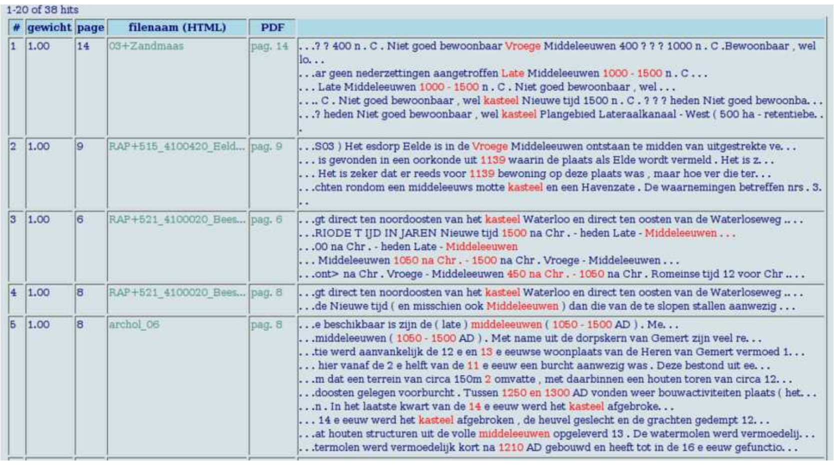
Strings in the text that are relevant to the query are in red. Strings that are not relevant, but that are recognized as geographical or chronological expressions are in blue.

You will observe that the HTML-file is not always well-aligned with the original 'image' of the page or that ugly overlaps or jumps in the text are visible. This cannot be helped without major surgery, and precisely for that reason we make the pdf-file also available to you. But in most cases the problems with rendering are minor or not even visible.

2.7 Other options

Apart from the different ways to weigh the keywords, there are some other options visible when you select “show options”.

- **Order by.** (Order by=weight/filename) Order results by weight (high to low) or by filename (alphabetically).



1-20 of 38 hits

#	gewicht	page	filenaam (HTML)	PDF	
1	1.00	14	03+Zandmaas	pag. 14	. . .? ? 400 n . C . Niet goed bewoonbaar Vroege Middeleeuwen 400 ? ? ? 1000 n . C .Bewoonbaar , wel lo. . . . . .ar geen nederzettingen aangetroffen Late Middeleeuwen 1000 - 1500 n . C . . . . . . Late Middeleeuwen 1000 - 1500 n . C . Niet goed bewoonbaar , wel . . . . C . Niet goed bewoonbaar , wel kasteel Nieuwe tijd 1500 n . C . ? ? ? heden Niet goed bewoonba. . . . . .? heden Niet goed bewoonbaar , wel kasteel Plangebied Lateraalkanaal - West (500 ha - retentiebe. . .
2	1.00	9	RAP+515_4100420_Eeld...	pag. 9	. . .S03) Het esdorp Eelde is in de Vroege Middeleeuwen ontstaan te midden van uitgestrekte ve. . . . . . is gevonden in een oorkonde uit 1139 waarin de plaats als Elde wordt vermeld . Het is z. . . . . . Het is zeker dat er reeds voor 1139 bewoning op deze plaats was , maar hoe ver die ter. . . . . .chten rondom een middeleeuws motte kasteel en een Havenzate . De waarnemingen betreffen nrs . 3. . .
3	1.00	6	RAP+521_4100020_Bees...	pag. 6	. . .gt direct ten noordoosten van het kasteel Waterloo en direct ten oosten van de Waterloseweg .. . . . . .RIODE T IJD IN JAREN Nieuwe tijd 1500 na Chr . - heden Late - Middeleeuwen . . . . . .00 na Chr . - heden Late - Middeleeuwen . . . Middeleeuwen 1050 na Chr . - 1500 na Chr . Vroege - Middeleeuwen . . . . . .ont> na Chr . Vroege - Middeleeuwen 450 na Chr . - 1050 na Chr . Romeinse tijd 12 voor Chr .. . .
4	1.00	8	RAP+521_4100020_Bees...	pag. 8	. . .gt direct ten noordoosten van het kasteel Waterloo en direct ten oosten van de Waterloseweg .. . . . . .de Nieuwe tijd (en misschien ook Middeleeuwen) dan die van de te slopen stallen aanwezig . . .
5	1.00	8	archol_06	pag. 8	. . .e beschikbaar is zijn de (late) middeleeuwen (1050 - 1500 AD) . Me. . . . . .middeleeuwen (1050 - 1500 AD) . Met name uit de dorpskern van Gemert zijn veel re. . . . . .tie werd aanvankelijk de 12 e en 13 e eeuwse woonplaats van de Heren van Gemert vermoed 1. . . . . . hier vanaf de 2 e helft van de 11 e eeuw een burcht aanwezig was . Deze bestond uit ee. . . . . .m dat een terrein van circa 150m 2 omvatte , met daarbinnen een houten toren van circa 12. . . . . .doosten gelegen voorburcht . Tussen 1250 en 1300 AD vonden weer bouwactiviteiten plaats (het. . . . . .n . In het laatste kwart van de 14 e eeuw werd het kasteel afgebroke. . . . . . 14 e eeuw werd het kasteel afgebroken , de heuvel geslecht en de grachten gedempt 12. . . . . .at houten structuren uit de volle middeleeuwen opgeleverd 13 . De watermolen werd vermoedelij. . . . . .termolen werd vermoedelijk kort na 1210 AD gebouwd en heeft tot in de 16 e eeuw gefunctio. . .

Figure 3: Display in context

- **Max. hits**. (maxhits=number) Number of hits shown on a single page in the list with results.
- **Show chronological graph**. (chron_graph=0/1) Activating this option will cause a histogram to be displayed, with the frequency of the individual years in the pages found (see picture 4). Periods are expanded, so that 'middle ages' will cause all years between 500 - 1500 to be incremented by one. In this particular database interest seems to center on the years between the beginning of the iron age in Holland and the end of the middle ages. You will observe the very human tendency to gravitate towards 'round' years, such as 500 or 1000.
- **Show map with city/village names**. (place_graph=0/1) Activating this option will display a map with the city and village names found by your query. (Still in experimental fase).
- **Show doclist for geo-information**. (show_doclist=0/1) This option will show a list of documents voor geographical information.
- **Show hits in context**. (kwic_display=0/1). Not to be confused with the KWIC search above. kwic_display causes the context of keyword, chronological and geographical queries to be shown in its context before you jump to the page itself. See figure 3).
- **Expert mode**. (expert=0/1) . In expert mode you are presented with just one space to enter your questions. Options et cetera are set by adding expressions like 'option=value'; e.g. 'expert=1'. Refer to table 2.7.
- **Interface language**. (language=EN/NL) This option lets you choose between displaying the interface in either English or Dutch.

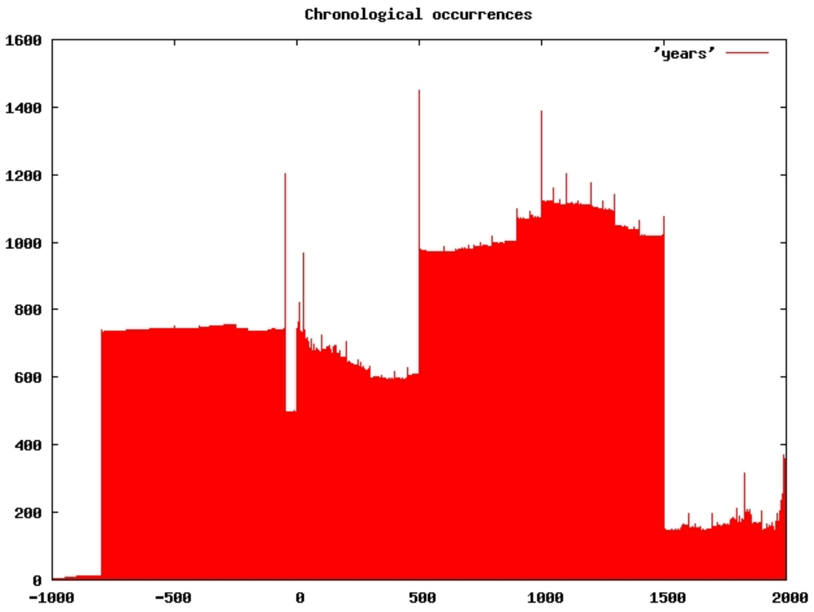


Figure 4: Histogram of chronological references between 1000 BC and 2000 AD

- **discard all search strings**. (discard_searchstrings=0/1) This option gives you the option to discard all search strings.
- **Change database**. (database=name/¡empty¿) Returns you to the first page of Open Boek with a different database (or none).
- **Administration**. This option is to be found at the bottom of the page, seperately from the other options. Starts the administration interface for the creation of new databases, indexing and similar activities.

2.8 Moving databases

To backup and restore databases you will need the administration password and acces http://yourserver/theopenboekdirectory/Programs/admin.php. Familiarize yourself with the menus from figure 10 and 12.

In the case that you want to move a complete, indexed database from one computer to another, please note the following: If you copy the complete Openboek installation, you need to update the 'openboek.rc' file, so that the program dir, the datadir and the writedir are all defined. If you move normal databases, see to it that they are placed in the datadir, and that the corresponding mysql diorectories and the entry in Openboek.db_lijst are also copied. Also, if the pdf-files are dynamic links, please

options=1/0	show options
expert=1/0	expert mode on/off
biblio=1/0	show bibliography
kwic_display=1/0	context on/off
chron_graph=1/0	show chronological histogram
place_graph=1/0	show found place names on map
debug_table=1/0	show debug table
maxhits=n	number of hits per table
database=naam	new database
order=docnaam/	order by name document (ascending alphabetically)
order=gewicht/	order by weight (descending)
order=startpag	order by order in database (ascending)

Table 1: Expert commands

1.	`cat dog horse`
2.	`1200-1400`
3.	`@1200-1400`
4.	`1200-1400; cat dog horse`
5.	`middle ages - second worldwar; cat dog horse`
6.	`1200BC-1000BC`
7.	`amsterdam(20)`
8.	`1200-1400; cat dog horse amsterdam(20)`
9.	`den_bosch`

Table 2: Valid queries in Open Boek

ensure that the link is accessible from the new directory. Perhaps it is easier to backup to a CD or usb stick as described below, because it takes care of all these details for you.

2.8.1 Backup to CD

You can burn a CD with the contents of one or more Database directories or copy them to an usb stick. To do this, make sure that you can read from and write to the corresponding mysql directories, e.g.: /var/lib/mysql. For this, add mysql and www-data to the group users and change the mysql directories to that group. Do not forget to change the permissions to g+rwx, or if you only make a backup, at least to g+r.

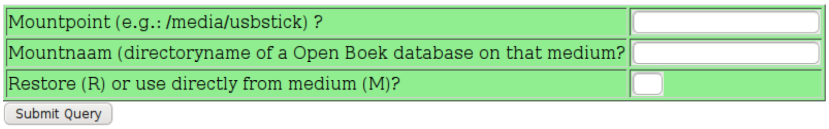
To export a database, go to the administration menu for that database (figure 10) and select the 'export' button. Now, all the relevant data will be written to the normal database directory. These are a small file 'filelist' and the mysql directory for your database.

Then copy the complete directory to the usbstick. Use a graphical browser such as Dolphin or bite the bullet and use the bash shell.

Mount the medium. Then use the command `cp -RL source destination` to ensure that both internal directories (R) and linked files (L) are copied. If you burn to a CD, also ensure the copying of the linked files, using the options of your burning program.

Now you can restore your data from the CD, or alternatively access the database directly from the CD, without copying its contents.

2.8.2 restoring from CD

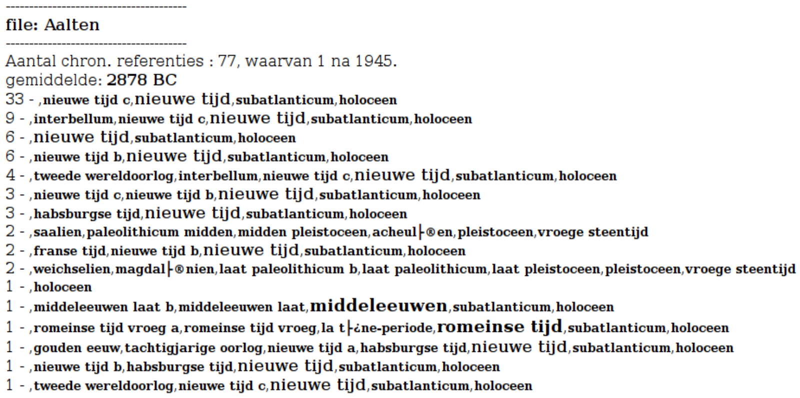


Mountpoint (e.g.: /media/usbstick) ?	
Mountnaam (directoryname of a Open Boek database on that medium?	
Restore (R) or use directly from medium (M)?	

Figure 5: Importing a database

Of course, you must have a functional Open Boek on your PC before you can backup from CD or usbstick, or consult the data directly from the CD. Check if the users and permissions for the mysql directory are set as described above and mount your CD or usb stick.

Choose 'import' from the administration menu. You will be presented with three questions: where the CD is mounted, the name of the database and whether you want to restore the data or use it directly from the mounted medium.



```
--------------------------------------
file: Aalten
--------------------------------------
Aantal chron. referenties : 77, waarvan 1 na 1945.
gemiddelde: 2878 BC
33 - ,nieuwe tijd c,nieuwe tijd,subatlanticum,holoceen
9 - ,interbellum,nieuwe tijd c,nieuwe tijd,subatlanticum,holoceen
6 - ,nieuwe tijd,subatlanticum,holoceen
6 - ,nieuwe tijd b,nieuwe tijd,subatlanticum,holoceen
4 - ,tweede wereldoorlog,interbellum,nieuwe tijd c,nieuwe tijd,subatlanticum,holoceen
3 - ,nieuwe tijd c,nieuwe tijd b,nieuwe tijd,subatlanticum,holoceen
3 - ,habsburgse tijd,nieuwe tijd,subatlanticum,holoceen
2 - ,saalien,paleolithicum midden,midden pleistoceen,acheul├®en,pleistoceen,vroege steentijd
2 - ,franse tijd,nieuwe tijd b,nieuwe tijd,subatlanticum,holoceen
2 - ,weichselien,magdal├®nien,laat paleolithicum b,laat paleolithicum,laat pleistoceen,pleistoceen,vroege steentijd
1 - ,holoceen
1 - ,middeleeuwen laat b,middeleeuwen laat,middeleeuwen,subatlanticum,holoceen
1 - ,romeinse tijd vroeg a,romeinse tijd vroeg,la t├¿ne-periode,romeinse tijd,subatlanticum,holoceen
1 - ,gouden eeuw,tachtigjarige oorlog,nieuwe tijd a,habsburgse tijd,nieuwe tijd,subatlanticum,holoceen
1 - ,nieuwe tijd b,habsburgse tijd,nieuwe tijd,subatlanticum,holoceen
1 - ,tweede wereldoorlog,nieuwe tijd c,nieuwe tijd,subatlanticum,holoceen
```

Figure 6: Chronological metadata

2.9 Chronological metadata

If you have a collection of documents indexed chronologically, you can easily generate a list of the chronological data for every document. Again, go to the administration menu for your database (figure 12) and select 'collect chronological metadata'. After a few seconds or minutes, you will get a long list looking like the figure 6.

The chronological references in every file are translated back to the eras in eras.all.rc, including all comprising eras and sorted on descending order. So a reference to the year 1942 will be expanded to “tweede wereldoorlog,interbellum,nieuwe tijd c,nieuwe tijd,subatlanticum,holoceen”, because the eras.all.rc file describes it as belonging to all these eras.

Then, because many eras include the name of broader eras (middle ages - late middle ages), the names get more or less emphasis when they are included often or less often.

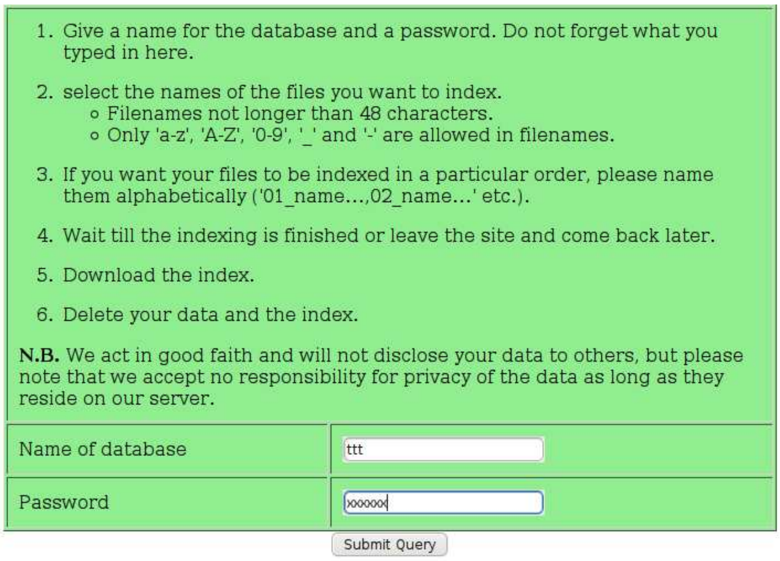
2.10 Using the indexing service

Open Boek includes an indexing server in which you can upload pdf-documents. The server will index the files chronologically and put the indices at your disposal for inspection or downloading. You can access the server as .../Programs/server.php.

In figure 14 you are invited to choose a name for your database and a password. The password is needed for getting the completed indices and to remove them after downloading or when you do not need them any more.

The second screen (figure 15allows you to name the pdf-files that you want to upload. Note that there is a upper limit to the combined size of 500 Mb for the files to be uploaded.

After confirmation, you will see Open Boek getting in action. You can



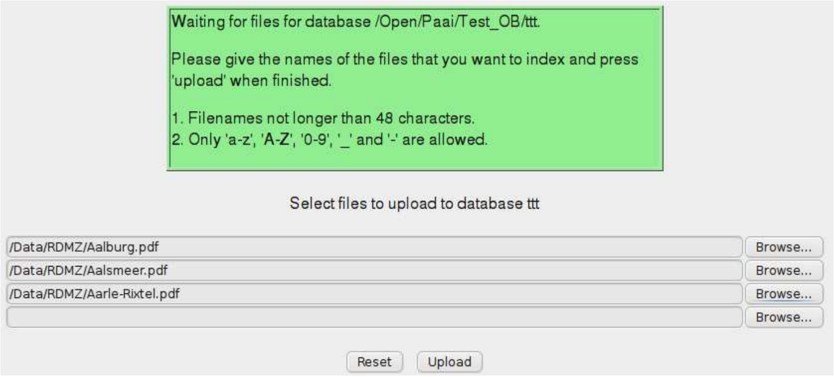
1. Give a name for the database and a password. Do not forget what you typed in here.
2. select the names of the files you want to index.
   - Filenames not longer than 48 characters.
   - Only 'a-z', 'A-Z', '0-9', '_' and '-' are allowed in filenames.
3. If you want your files to be indexed in a particular order, please name them alphabetically ('01_name...,02_name...' etc.).
4. Wait till the indexing is finished or leave the site and come back later.
5. Download the index.
6. Delete your data and the index.

N.B. We act in good faith and will not disclose your data to others, but please note that we accept no responsibility for privacy of the data as long as they reside on our server.

Name of database	ttt
Password	xxxxxx

Figure 7: The indexing server (1)

leave the browser and retrurn later to .../Programs/server.php with the name of your database and your password, to see if the job is finished. If it is, you can inspect or download the results.



Waiting for files for database /Open/Paai/Test_OB/ttt.

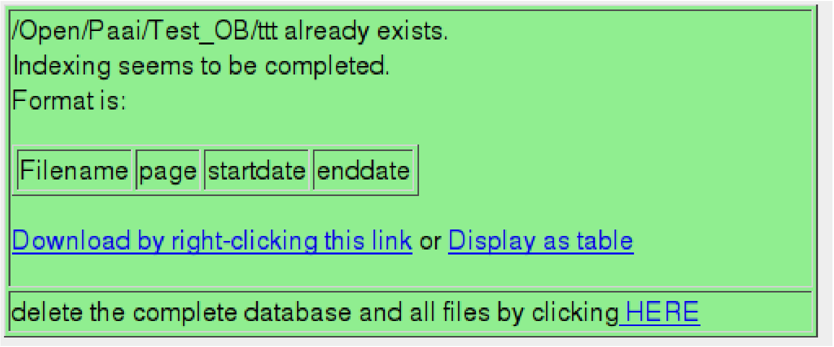
Please give the names of the files that you want to index and press 'upload' when finished.

1. Filenames not longer than 48 characters.
2. Only 'a-z', 'A-Z', '0-9', '_' and '-' are allowed.

Select files to upload to database ttt

/Data/RDMZ/Aalburg.pdf
/Data/RDMZ/Aalsmeer.pdf
/Data/RDMZ/Aarle-Rixtel.pdf

Figure 8: The indexing server (2)



/Open/Paai/Test_OB/ttt already exists.
Indexing seems to be completed.
Format is:

Filename	page	startdate	enddate

Download by right-clicking this link or Display as table

delete the complete database and all files by clicking HERE

Figure 9: The indexing server (3)

3 Installation

This section covers the installation of Open Boek and will let you create your first indexes, so that you get a feeling of the system and its administration. A detailed overview of Open Boek administration is to be found in chapters 3.2 and 4.

Open Boek runs as a collection of scripts under a http server such as Apache. For these scripts and the infrastructure you must have a Linux system available, because the Microsoft environment does not support all the necessary tools. The administrator of the system should have some elementary knowledge of Unix systems, know how to install new software, use the command line interface and have the authority to change permissions. It is possible that some of the third party software has to be (re-)compiled.

We will describe in detail the steps that will be necessary to index the files in the Database-directories. There is a web-interface available (admin.php) with as URL http://.../admin.php. You will need a password to enter this URL: for the moment this is 'admin'. If you want to change it, you will have to do this in the source of admin.php.

Requirements, software

The software requirements of Open Boek are:

- a modern Unix system, such as Linux, including Apache, MySQL and PHP. We used SuSE and later Ubuntu. Some Linux distributions have non-standard versions of awk or the shell (Ubuntu); we assume (g)awk and bash.
  Please note that you may have to increase the amount of memory that is allowed for PHP; adjust /etc/php5/apache/php.ini if necessary.
- the system files of Open Boek, available as a compressed tar archive[2].
- the pdf to html convertor, *pdftohtml* version 0.36[3].
- a program to split a large pdf in its separate pages: pdftk[4].
- a plotting program: gnuplot.
- a compiled version of the venerable [6] SMART retrieval system, version 11.0[5] from 1993. A linux binary can be found in the openboek.tgz file [3]; a clean compilation is not for the faint-hearted. Please copy the smart-binary and the smprint-script (tcsh) to /usr/local/bin. In a next version of Open Boek we may distribute an alternative indexing and retrieval engine.
- a version of TiMBL 6.1.5 [1] [6]. You need Timbl for chronological and geographical indexing, see chapter 3.2.
- a version of the Mbt 3.1.3 (Memory Based Tagger, also from Ilk). Needed for POS-tagging and geographical indexing.
- the Uber Uploader package[7] if you want to use the server-option with wich users can upload and index pdf-files.

Requirements, Data

Apart from these programs, Open Boek also needs some data and tables. We already included databases for use with Timbl and Mbt. See chapter 3.2 for detailed instructions. For the first installation and indexing as described in this chapter, they are not needed.

[2] http://www.referentiecollectie.nl/Openboek/openboek.tgz

[3] http://pdftohtml.sourceforge.net

[4] http://www.accesspdf.com/pdftk

[5] ftp://ftp.cs.cornell.edu/pub/smart/smart.11.0.tar.z

[6] The source is available at http://ilk.uvt.nl, but you will have to compile it yourself.

[7] http://uber-uploader.sourceforge.net/

Important: All directories in use by Open Boek, including the Database-directories and the mysql-directories, should be read-, write- and executable for your http-server. On a Linux system the http-server will generally be user *wwwrun* or *www-data*. You can also assign a group, e.g.: 'users' that wwwrun, mysql and yourself belong to, and set the permissions u+rwx for those directories, so that you can inspect and change scripts from the command line, if and when needed.

3.1 Install and prepare Open Boek

The openboek.rc file

After unpacking and checking the availability of the programs on which Open Boek depends, first edit the *openboek.rc* file. This file is a small text file with some data that Open Boek should know about (see table 3). Essentially those are the name of the server and the location of the Open Boek programs and scripts. Other things, such as preferred language for the interface are also changed here, but for most variables reasonable defaults exist. Note that the hashmark (#) precedes comments, that are not interpreted by the system.

In the example (table 3) we assume that your documentroot according to apache is /Open and that you unpacked your openboek.tgz file in /Open/Openboek/Stable, creating the directory Programs in the process.

The password for the SQL user is coded. Use your browser to access http://.../Programs/admin.php?cmd=codepas where you can enter your sql-password. It returns a string that should be copied to the openboek.rc file.

If you want to use a different language for the interface, add a variable *interface_lang*. Dutch (default) is 'NL', english is 'EN'. Other languages can be added, but you should create and edit separate dialogs- and help-files in that language. If you want to add, e.g. german, you would choose 'DE' as the value of *lang* and create the 'dialogs.DE' and 'help.DE.html' files as translations of their dutch and english counterparts.

Nota Bene: the language of the interface is *not* necessarily the language of the documents in the database. If you want to add a database with documents in a language other than dutch, please refer to subsection 4.6.

Now, go to the Programs directory and check the dynamic link index.php, which should be a dynamic link to engine.php.

If you want to use Googlemaps, you must register the server with Google and add te key in the Coords-directory (coords.php and coords2.php).

If you use SMART and/or TIMBL, see to it that you have read the license agreements, and have installed the binaries in /usr/local/bin. The same is true for pdftohtml, pdftk and all other programs that you use. To my best knowledge all programs that Open Boek uses, are free for non-commercial purposes, but do not take anything for granted.

```
# open boek directory minus the 'document root' /Open
wwwdir=/Openboek/Stable

# complete directory open boek
rootdir=/Open

# program directory
programdir=/Open/Openboek/Stable/Programs

# data directory (not writable after indexing is done)
datadir=/Open/Paai/OB

# write directory (always writable for logfiles and intermediary stuff
writedir=/Open/Openboek/Stable/Writable

# your hostname
hostnaam=http://www.myhostname.nl

# preferred language of the interface
lang=EN

# SQL user
sql_user=root

# SQL password (coded)
sql_password=xyz...
```

Table 3: The openboek.rc file

3.2 Creating and indexing a database.

Again note that the home directory and all directories under it should have rwx-permissions for the http server.

Step 1: select the documents

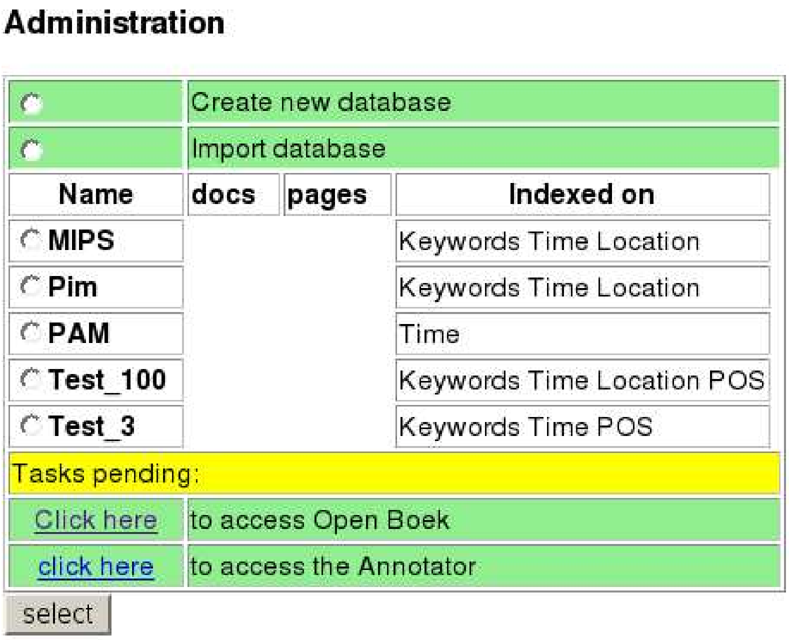
With an ASCII editor, create a list of the pdf-files or html-files you want to include in your database, with complete path information. It is a good idea to move this file to your Open Boek home directory and keep it there. You can mix html and text files with the pdf-files, but the .html and .txt suffixes are obligatory.

Important. The inclusion of html and text are not yet tested in version 3.0 of Open Boek.

Step 2: prepare the database

Open de URL http://.../Programs/admin.php (protected with password) and select the uppermost option (Create new database)(see 10).

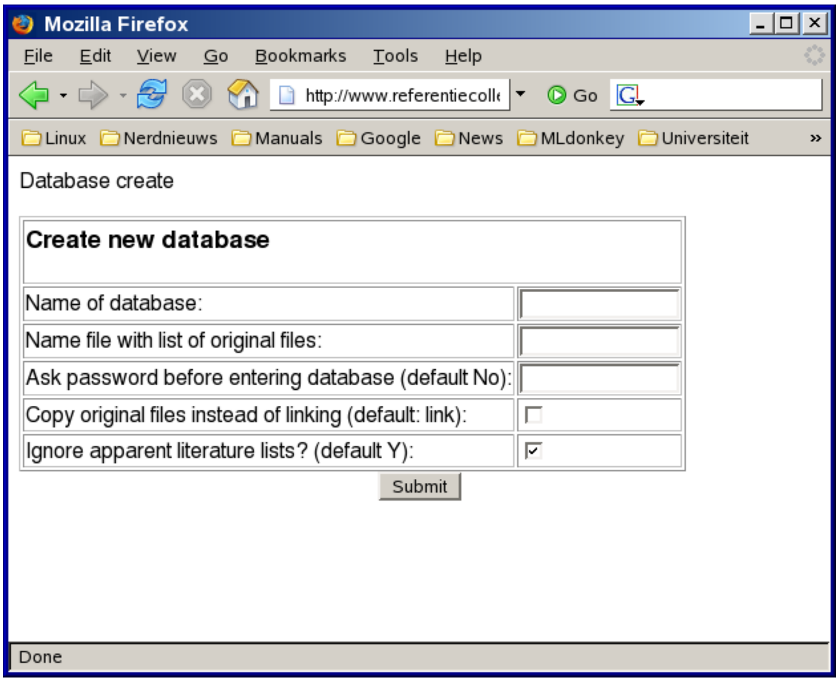
A new screen is displayed (11): give a name for the new database that starts with a capital and provide the name of the file with pdf-files relative to the home directory of OB. There are a few more options parameters that you should know about.



Administration

○	Create new database		
○	Import database		
Name	docs	pages	Indexed on
○ MIPS			Keywords Time Location
○ Pim			Keywords Time Location
○ PAM			Time
○ Test_100			Keywords Time Location POS
○ Test_3			Keywords Time POS
Tasks pending:			
Click here	to access Open Boek		
click here	to access the Annotator		

Figure 10: Menu 1 for database administration



Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://www.referentiecolle
Go
Linux Nerdnieuws Manuals Google News MLdonkey Universiteit
Done

Database create

Create new database

Name of database:	
Name file with list of original files:	
Ask password before entering database (default No):	
Copy original files instead of linking (default: link):	□
Ignore apparent literature lists? (default Y):	☑

Figure 11: Creating a new database

```
# password
passwd=apekool

# ignore pages that look like bibliographies (Y or empty)
ignore_lit=Y

# local options (Y or empty)
local_options=

# display illustrations (Y or empty)
ill_zichtbaar=Y

# language of the database (NOT language of the interface)
LANG=NL
```

Table 4: The database.rc file

The first is whether you want the pdf files copied to the Open Boek structure, or just have them linked there. The default is linking; but if you want to burn your database on a CD or DVD, you must copy the original pdf-files to the OB structure.

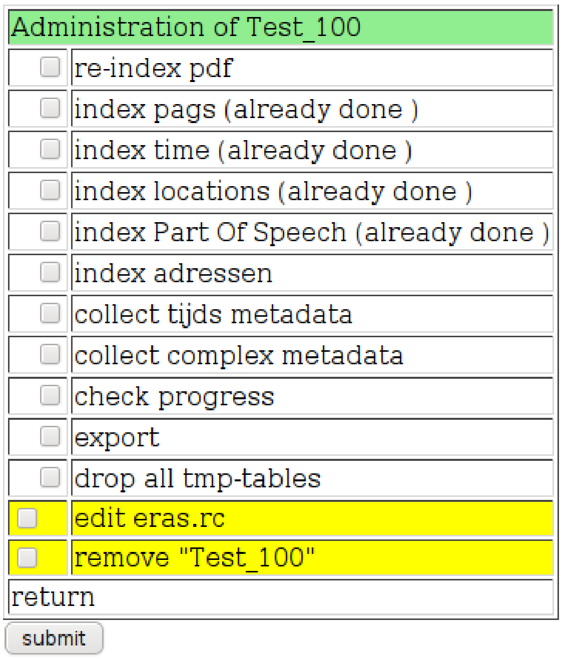
The second is whether Open Book will try to recognize pages that contain literature references. Such references almost always contain place names and years, but such data are mostly 'uninteresting' as search argument. For instance: many dutch archeological texts are published in Amersfoort; and such occurrences will strongly interfere with a search for archeological finds in or round Amersfoort. The default is therefore to ignore literature lists. In chapter 4 you will find instructions on how to adjust the rules under which bibliographies and tables are recognized.

Now press 'submit'. A new directory with the name of your database is created (.../Database) and the pdf files from the list will be copied (or linked) to their appropriate sub-directories under Database/Data. This can take some minutes for very long lists, but you can close the browser or surf to another URL if you want to.

There will also be a new directory with the name .../Writable/Database with logfiles and pid-files. If something goes awry during preparation or indexing, you will have to remove the pid-file (in this case classify_prepare.pid) by hand. If you remove the pid-file while the system is busy, Open Boek will stop.

When this stage is finished, you can reload the page with the administrative interface (../admin.php). Your new database should now be visible. Select it, so that the menu in figure 12 is displayed. you will observe that the actions you can perform on every database are governed with buttons. Every database can have different indexes activated: this is indicated in the last column.

If indexing is in progess, you will be notified by the fact that this is indicated in red. Also, in the yellow box at the bottom, the tasks that are currently running, are displayed. It is generally a good idea not to start



Administration of Test_100

- □ re-index pdf
- □ index pags (already done)
- □ index time (already done)
- □ index locations (already done)
- □ index Part Of Speech (already done)
- □ index adressen
- □ collect tijds metadata
- □ collect complex metadata
- □ check progress
- □ export
- □ drop all tmp-tables
- □ edit eras.rc
- □ remove "Test_100"

return

submit

Figure 12: Menu for individual database administration

new tasks when this is in evidence.

Step 3: indexing, keywords

You may want to add or edit the 'stopw.*' files in the Database-dir. These files contain lists of commonly used words that will not be indexed. The keyword indices will be prepared by SMART. This task only takes a few seconds, (longer for large databases) after which you can use the advanced keyword search features.

Nota Bene: All actions in Open Boek leave logfiles in the Writable/Database directory. See the section 5 on the names of the logfiles and when they are created. At this stage of development, the logfiles are overly verbose.

Nota Bene: When index activities take place, a corresponding 'pid' file is used to prevent the same indexing being started twice. If something goes wrong during indexing, the pid-file is not removed, and you will have to delete it by hand from .../Writable/Database

This ends the instructions on how to create and index an Open Boek database.

The following sections cover the creation of chronological and geographical indexes and the tools needed for them.

3.3 chronological indexing

For the indexing of chronological expressions, you need a file with tagged examples for the language of your database. The same is true for the creation of geographical indices. These files reside in the directory *Programs/General* of Open Boek. The file with tagged chronological examples for dutch is called ’chron_examples.ann.NL’ and is included. The files for english and german are also included, and are called ’chron_examples.ann.EN and ...DE respectively. For maximum performance, create your own annotated files.

These files are coded, and are created from chron_examples.ann.NL.org (dutch). Do note that after annotating a file yourself, you have to translate the file with *’General/translate_examples.awk’* (also see section 7).

To index chronologically, start the administration tool, and select your database as described in chapter 3. Then, select the appropriate checkbox for the action you want to perform (see fig. 11). Very large databases with thousands of documents can easily take two or three days to complete the indexing, but you can leave the browser at any moment. If you want to return and check how far your indexing has progressed, choose *check progress* from the menu and you wil get a report on the operations going on for that database.

3.3.1 The eras.all.rc file

An important file for chronological indexing is the eras.rc file. It contains the definitions of chronological names, such as the middle ages or the paleolithic in three languages (dutch, english and german). A default eras.rc is included in the distribution, we suggest that you adjust it for your own applications. If you use variations or additions such as ’late middle ages’, see to it that the longer names precede the shorter names in the eras.rc file. The separators between the columns are the ’—’ sign. Fields may be empty, except the two last fields.

When creating the original databases, you had the option of ignoring apparent literature. If this option is enabled (to ignore literature), Open Boek tries to guess which years are part of a bibliographic reference. Such years are omitted from the index, although they are marked in the HTML display.

Midden Mesolithicum	middle mesolithic	Mittleres steinzeit	-7100	-6450
laat Mesolithicum			-6450	-4900
Late steentijd	Late stone age	Spaete steinzeit	-5300	-2000

Table 5: The eras.all.rc file

```
use Openboek;
drop table geonames;\index{geography}

create table geonames\index{geography} (geonameid int auto_increment primary key,
name varchar(200),
asciiname varchar(200),
alternatenames varchar(4000),
latitude float,
longitude float,
feature_class varchar(1),
feature varchar(10),
country varchar(2),
cc2 varchar(60),
admin2\index{administration} varchar(80),
admin3\index{administration} varchar(20),
admin4\index{administration} varchar(20),
population int,
elevation int,
gtopo30 int,
timezone varchar(50),
modification date);

load data local infile "allCountries.txt" into table geonames;\index{geography}
update geonames\index{geography} set asciiname=lower(asciiname);

alter table geonames\index{geography} add index\index{index} asciiname(name);
alter table geonames\index{geography} add index\index{index} latitude(latitude);
alter table geonames\index{geography} add index\index{index} longitude(longitude);
```

Table 6: To import the allCountries.txt file into Mysql

3.4 indexing, geographical

The first condition for the indexing of placenames is a database with all cities and villages of the world.

This database is found at http://www.geonames.org/export/. In the table 6 you will find the sql commands to read this table into MySQL. Note that the field 'asciiname' will be concerted to lowercase.

If you want to be able to retrieve locations that have a hyphen in them, you will have to hack your characterset in /usr/share/mysql/charsets. Too bad! Open, e.g. ascii.xml, locate in $< ctype >< map >$ the character '-' (number 45 or 0x2Dm which is on the third long line the third code from the right) and change it to '02'. Do this before indexing the table.

3.4.1 POS-tagging

Before geographical indexing can be started on a database, you need to do Part-Of-Speech tagging. POS-tagging is done by the program Mbt that should reside in /usr/local/bin. It uses example files in Programs/Tagger/NL (dutch), .../EN for english and .../DE for german.

NB: at the moment there are no databases for english or german examples. The files that reside in the repective directories are the dutch ones and only there for testing purposes!

Go to the administration menu and select the administration of the database that you want POS-tagged. Select the checkbox for POS-tagging.

3.5 Indexing of placenames

After the POS-tagging you can proceed to the indexing of places, Choose the appropriate checkbox from the menu.

3.6 Indexing of addresses

(experimental) Choose the appropriate checkbox from the menu. The result will be written to Database/Adressen/adressen

3.7 Some notes on document file formats

Open Boek supports pdf, text and html formats. If you start with other formats, convert them to either pdf or HTML, but note that you need a textual representation of your document in the pdf-file. Pdf's that only contain e.g, scanned pages without having been OCRred, will not be indexed.

Pdf is logically structured as paged documents, and OB will take those pages as units vor indexing and display.

HTML has no page structure. If you want to paginate HTML-files, insert the line <!- - pagina - -> (html comment) where you want your pagebreaks. OB will put <body>...</body> tags around the individual pages, otherwise it is your responsibility to see that the HTML within the pages always is consistent, that the tags are balanced etcetera.

Often, pdf documents were originally typed on paper, and later scanned, OCR-red and stored as PDF. In such files, the 'image' of every page is paired by an 'invisible' ASCII text that however can be easily extracted and indexed. The problem here is the display of the retrieved pages. The original pdf-images of course contain all sorts of pictures, tables and drawings, but we did not address the technical problem of highlighting keywords or the addition of links in that pdf-representation. Instead we convert the contents to HTML. However: this gave rise to the following problems.

1. One alternative, the omission of the image of the page, and the display of only the ASCII text as HTML gave the opportunity of highlighting and links, but omitted most visual content such as images and most formatting.

2. The second option consisted of the projection of the HTML-ized ASCII over the image. This combines highlighting, links *and* visual content, but the result in the browser often looks messy.

Another large portion of the files was already written using a wordprocessor and stored as PDF. Such files translated relatively easy in HTML, combining highlighting, links and images. Still, the rendering of the fonts is not always satisfactory. In any case you can switch from one method of display to the other during display of the page.

The default in Open Boek is (1). If you want to change the default, In the database directory Database exists a file 'Database.rc'. In this file, you can put the line `ill_zichtbaar=Y`. In that case, the default will be that the illustrations are visible.

Tables and other artefacts

One of the problems with the conversion program that we used is that the resulting HTML is divided in lines (in the sense of one or more words on the same level), and that every such line is only marked by its position on the page and its font. Subscripts and superscrips are not considered part of the line; they get individual tags for font and position, after which a new 'line' is started. Every information about e.g. the line being part of a table, header or caption, is lost. A similar problem exists if the text is made up in columns; our programs do not recognize the columns but read the two lines as belonging to a single line. These problems are not solved at this moment.

Microsoft files

A third group of documents consisted of hundreds of reports written by individual archeological bureaus. These were stored on as many CDs and almost always produced by Microsoft software. Without a doubt every CD contains a highly artistic multimedia feast with sounds, movies and everything, but it was absolutely impossible to extract the original reports without a timeconsuming process of analysing the contents by hand, defeating the purpose of *automated* indexing and retrieval. But even if the 'central' document could be identified, Microsofts OLE framework often prevented extraction of the relevant data, at least with the tools that we used.

Another unexpected result of the Microsoft way of doing things was that we often found text or pictures in a Word file that were normally not visible, and certainly not meant to be visible, such as corrections, annotations and remarks, deleted pictures and so on. This can lead to embarrasing situations.

Sometimes Doc documents can be converted to pdf. Such pdf-files *can* be indexed normally.

4 Administration

This section describes the working of Open Boek in detail with emphasis on the files that are created and the directory structures that support it. For notes on individual programs refer to chapter 5.

Open Boek runs as a collection of scripts under a http server such as Apache. For these scripts and the infrastructure you must have a Linux system available, because the Microsoft environment does not support all necessary tools. The administrator of the system should have some elementary knowledge of Unix systems, know how to install new software, use the command line interface and have the authority to change permissions. It is possible that some of the third party software has to be (re-)compiled.

```
# password
passwd=apekool

# ignore pages that look like bibliographies (Y or empty)
ignore_lit=Y

# local options (Y or empty)
local_options=

# display illustrations (Y or empty)
ill_zichtbaar=Y

# language of the database (NOT language of the interface)
LANG=NL

# test option 1:  (if 1 omit administration of indexing)
test_option_1=0
```

Table 7: The database.rc file

4.1 Creating a new database

In chapter 3 you have seen how to create a new database and how to index it. In that chapter, we also displayed the general 'openboek.rc' file.

When Open Boek collected the documents from your document list to create a new database, it performed the following actions:

- An entry for the database is added to the table db_lijst in the general Openboek SQL database (refer to chapter 6).
- A separate SQL database is created with the name of the database and the table 'filelijst' (also refer to chapter 6).
- A number of specification files for SMART are copied from the /Programs/general directory to the new Database-dir, including lists with stopwords for different languages (stopw.NL etcetera).

- The file 'eras.all.rc' is copied to the Database directory. This file contains named chronological periods in Dutch, German and English. (see table 8). You should edit it according to your needs. Of course you can add new periods at will, as long as you conform to the examples:
  "naam_periodeNL|naam_periodeDU|naam_periodeEN|begin|einde"
  Years before christ are preceeded by a minus sign.

```
Midden Pleistoceen|Middle Pleistocene|Mittleres Pleistozn|-850000|-1280
Vroeg Pleistoceen|Early Pleistocene|Unteres Pleistozn|-2588000|-850000
Midden Bronstijd B|Middle Bronze Age B|Mittlere Bronzezeit B|-1500|-1100
Vroege Bronstijd|Early Bronze Age|Frühe Bronzezeit|-2000|-1800
...
```

Table 8: The eras.rc file

- A conversion from pdf to html is performed. In this step, which may take some time (approx. three or four pdf-files in a minute) OB will convert the contents of the pdf-files to HTML, images (png-files) and other relevant material, notably the token-lists and the taglists: one for the tokens of the text proper (doc-x_tokens), one for the interpunction (doc-x_interp) and one for the HTML tags (doc-x_taglist). The 'x' in the filename stands for the pagenumber. From now on, Open Boek will use these files to reconstruct the html-files at query time, and the original HTML-file can be discarded. In the doc-x_tokens-file every token is stored on a line of its own; in the doc-x_taglijst and later in the chronological and other tag-files, every tag is preceded by a number that refers to the linenumbers of this doc-x_tokens file.

- A file 'Database.rc' is created in the Database directory. This file is an extension of the 'openboek.rc' file so that variables specific to that database can be defined, e.g: 'filecopy' if you want to copy the files in stead of linking them. Here also the variable LANG is specified if the language of the documents is other than dutch (fig. 7).

- Open Boek makes an educated guess whether the document is dutch, englsh, german or french, and stores that information in the table filelist.

4.2 Files and Directories

If you have the system up and running and have created all indexes, you will find the following directories (we will call the directory where Open Boek was installed originally 'home') as depicted in fig. 13:

1. (home). Here the Programs directory, the Coords directory and the database directories are stored.

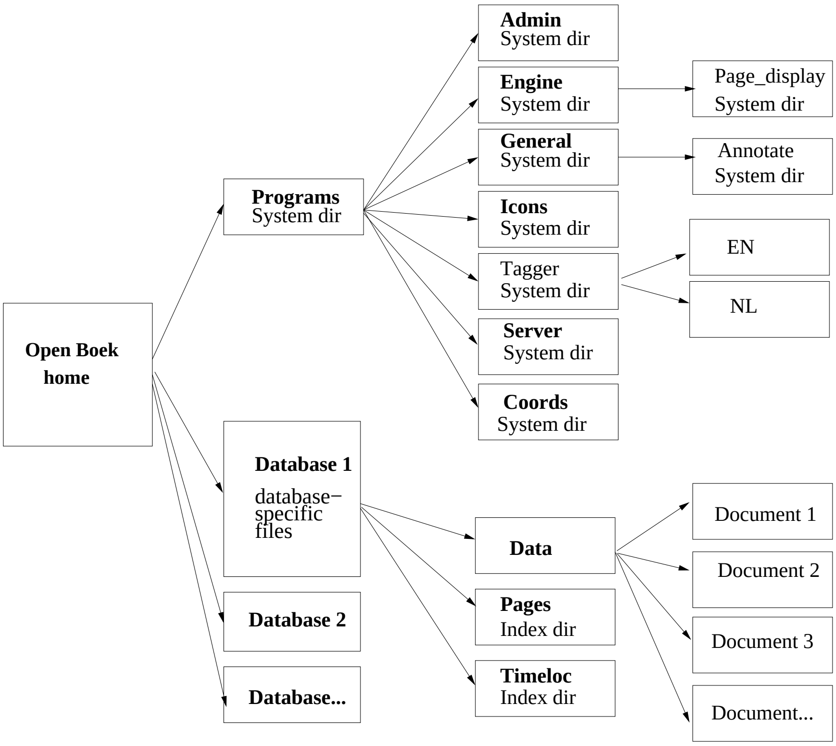


Figure 13: Dir structure

2. Coords. A directory with scripts to access Googlemaps for coordinates. For every directory with such scripts a separate license must be obtained from Google, although for the moment (2007) this is without cost.

3. Programs. As we said, the directory where the programs for Open Boek are stored: engine.php for the search and display, admin.php for the administration and server.php for remote indexing. The default "eras.rc" and the "openboek.rc" are stored here

   - Admin. Include files for admin.php
   - Engine. Include files for engine.php.
   - General. Dir for general files needed for Open Boek. Also contains the folder Annotate, a tool for annotating text files.
   - Icons. The directory where the icons and other images that the system needs, are stored. You can also find the style sheet for the interface here.
   - Server. Stuff for server.php, also the folder Swfupload is found here.
   - Tagger. datafiles for the Mbt tagger.

4. (Database). For every database there will be an individual directory with a corresponding name. We will use the generic name 'Database' for now. Here the files, specific for that individual database are kept. The most important is "Database.rc" where individual settings for

that database are stored. These directories can be burned to a CD or DVD after indexing, because nothing is written or changed after indexing.

After indexing the following subdirectories will exist in the Database-directory:

- Data. The location of the pdf-files and tag-files.
- Data/(Documents). A series of directories, each corresponding to a single document. The name of the directory is the name of the original document, without its extension. When we refer to a directory ’Document’, we mean one of those directories. Every document is split in pages (if and when possible) and every page is split in functional files: one for the tokens, one with tags for the layout, one with chronology tags and so on. Also, some php-files that combine those functional files into a coherent html file, and that govern navigation are copied from the directory Programs/Data-php and stored here.
- Pages. The SMART indexes for the individual pages.
- Timeloc. The directory with indexes to retrieve chronological and geographical data.

5. Writable. Under Writable, a directory is kept for every database, where intermediary files and logfiles are kept. This division is done to enable the data itself to reside on a read-only medium such as a CD.

Almost all of these files and directories will be created automatically, either when unpacking the Open Boek distribution or when creating and indexing a database of documents.

4.3 The index files

If keyword indexes are created, we find ’word_weights.atc’ and ’index’ in the Pages directories. For the MBL-generated indexes, we have the ’Timeloc’ directory, which contains the ’chronlijst’ index and the ’loclijst’ indexes. The ’chronlijst’ depends on the existence of the machine learning components TiMBL and a database with examples. In the Open Boek distribution such a database is included (’chron_examples.ann.NL’), but you are encouraged to create your own examples. Please note that the indexing of these numeric classes is very time-consuming, depending on your hardware this can take several minutes for every document.

The ’loclijst’ index tries to identify place names in the same way. It uses the file ’loc_examples.ann.NL’ for this purpose, in combination with the geonames database (see 3). By default, Open Boek will try to recognize literature references, and ignore place names in such cases. For this purpose it uses a rough heuristic, based on the ratio of interpunction, capitals and words. Please note that other pages can look like literature and be ignored. However, we found that place names on such pages generally were 'uninteresting' for the same reason that place names in literature lists are 'uninteresting'. In any case, you can always use plain keyword search to retrieve any string on such pages.

After indexing, all indexes are imported into corresponding SQL tables.

4.4 Moving databases

In the case that you want to move a complete, indexed database from one computer to another, please note the following: If you copy the complete Openboek installation, you need to update the 'openboek.rc' file, so that the program dir, the datadir and the writedir are all defined. If you move normal databases, see to it that they are placed in the datadir, and that the corresponding mysql diorectories and the entry in Openboek.db_lijst are also copied. Also, if the pdf-files are dynamic links, please ensure that the link is accessible from the new directory. Perhaps it is easier to backup to a CD or usb stick as described below, because it takes care of all thede details for you.

4.4.1 Backup to CD

You can burn a CD with the contents of one or more Database directories or copy them to an usb stick. To do this, make sure that you can read from and write to the corresponding mysql directories, e.g.: /var/lib/mysql. For this, add mysql and www-data to the group users and change the mysql directories to that group. Do not forget to change the permissions to g+rwx, or if you only make a backup, at least to g+r.

To export a database, go to the administration menu for that database and select the 'export' button. Now, all the relevant data will be written to the normal database directory. These are a small file 'filelist' and the mysql directory for your database.

Then copy the complete directory to the usbstick. Mount the medium as follows `mount -o uid=mysql -o gid=users -o umask=0` Then use the command `cp -RL -p source destination` to ensure that both internal directories (R) and linked files (L) are copied as well as the owner mysql. If you burn to a CD, also ensure the copying of the linked files.

Now you can restore your data from the CD, or alternatively access the database directly from the CD, without copying its contents.

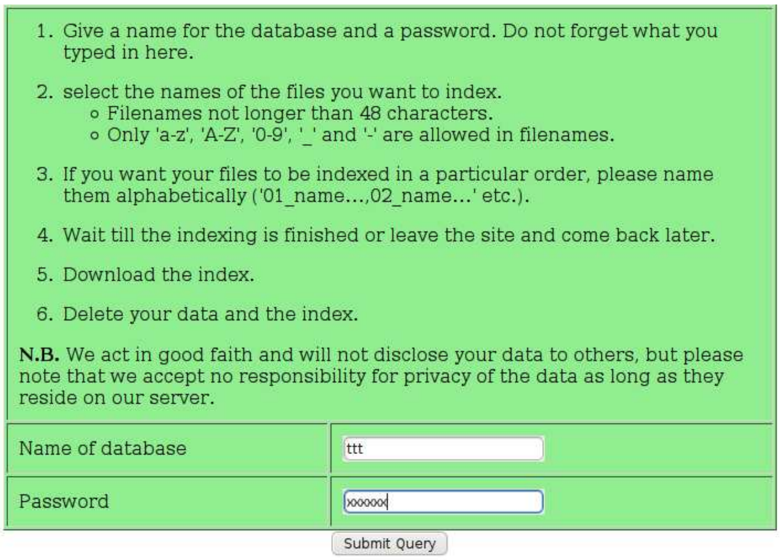
4.4.2 restoring from CD

Of course, you must have a functional Open Boek on your PC before you can backup from CD or usbstick, or consult the data directly from the CD. Check if the users and permissions for the mysql directory are set as described above and mount your CD or usb stick.

Choose 'import' from the administration menu. You will be presented with three questions: where the CD is mounted, the name of the database and whether you want to restore the data or use it directly from the mounted medium.

4.5 Using the indexing service

Open Boek includes an indexing server in which you can upload pdf-documents. The server will index the files chronologically and put the indices at your disposal for inspection or downloading. You can access the server as .../Programs/server.php.



1. Give a name for the database and a password. Do not forget what you typed in here.
2. select the names of the files you want to index.
   - Filenames not longer than 48 characters.
   - Only 'a-z', 'A-Z', '0-9', '_' and '-' are allowed in filenames.
3. If you want your files to be indexed in a particular order, please name them alphabetically ('01_name...,02_name...' etc.).
4. Wait till the indexing is finished or leave the site and come back later.
5. Download the index.
6. Delete your data and the index.

N.B. We act in good faith and will not disclose your data to others, but please note that we accept no responsibility for privacy of the data as long as they reside on our server.

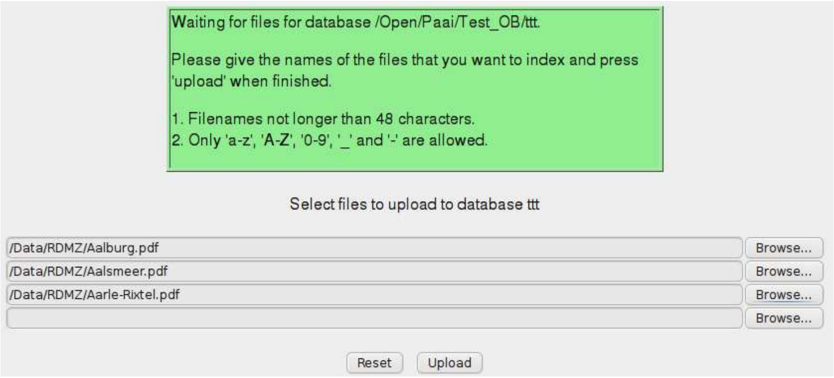
Name of database	ttt
Password	xxxxxx

Figure 14: The indexing server (1)

In figure 14 you are invited to choose a name for your database and a password. The password is needed for getting the completed indices and to remove them after downloading or when you do not need them any more.

The second screen (figure 15allows you to name the pdf-files that you want to upload. Note that there is a upper limit to the combined size of 500 Mb for the files to be uploaded.

After confirmation, you will see Open Boek getting in action. You can leave the browser and retrurn later to .../Programs/server.php with the name of your database and your password, to see if the job is finished. If it is, you can inspect or download the results.



Waiting for files for database /Open/Paai/Test_OB/ttt.

Please give the names of the files that you want to index and press 'upload' when finished.

1. Filenames not longer than 48 characters.
2. Only 'a-z', 'A-Z', '0-9', '_' and '-' are allowed.

Select files to upload to database ttt

/Data/RDMZ/Aalburg.pdf
/Data/RDMZ/Aalsmeer.pdf
/Data/RDMZ/Aarle-Rixtel.pdf

Figure 15: The indexing server (2)

4.6 Documents in other languages

Although you can easily change the language that is used for the interface (see subsection 3.1), it is more difficult to prepare Open Boek for documents in different languages.

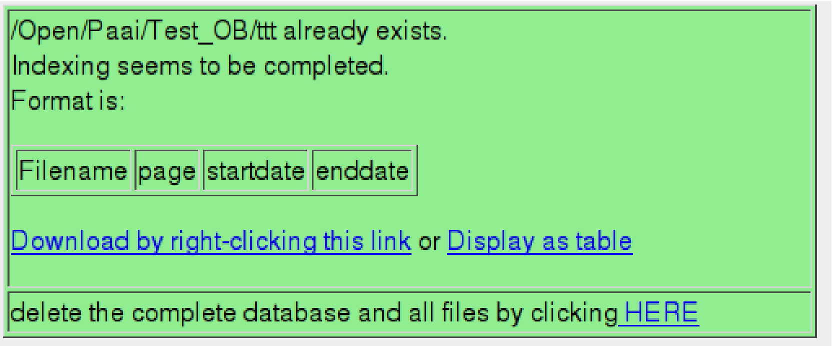
In any case, you should not mix languages in a single document. It is OK to have collections that consist of documents in different languages, but you will need parsers and example files for every individual language.

The system reads the database.rc file and extracts the parameter LANG. If this does not exist, dutch (NL) is assumed. This is the default for the collection. In the sql-table 'filelist', the educated guess of Open boek for every document is stored.

After that, you should realize that the language-dependent modules and files come in two groups: data and programs. They can be recognized by the NL, EN, DE etcetera infixes in the filenames.

One group those that select the cases for the machine learning part. Let us take the recognition and extraction of chronology data as an example. The modules that detects potential chronology-related phrases are paai_tag_time and num_pick. In these modules, we have three functions:

1. The first is to detect roman numerals and convert them to integers. This will not have to be replaced when you change from e.g., dutch to english. The source is in eval_roman.awk
2. The second function translates cardinals and ordinals to integers.



/Open/Paai/Test_OB/ttt already exists.
Indexing seems to be completed.
Format is:

Filename	page	startdate	enddate

Download by right-clicking this link or Display as table

delete the complete database and all files by clicking HERE

Figure 16: The indexing server (3)

Obviously this needs to be taylored to every language you want to use. Sources for english, german and dutch are in eval_cardinals_NL.awk, eval_cardinals_DU.awk and eval_cardinals_EN.awk.

3. Finally, there are some heuristics expressed as rules. These too are dependent on the particular language. They are implemented in paai_tag_time and num_pick themselves. **TODO:** take these heuristics add them to the language-dependent files.

The next thing to do is to extract about 10,000 examples of potential chronology-related phrases from a number of typical documents and categorize them by hand, possibly using the annotator described in section 7. The annotated lines are called time_examples.ann.LLorg with the language as suffix: e.g. chron_examples.ann.NL.org See table 26. Before they can be used, they should be converted.

There exists a dirty trick to extract such lines from the databases. What you do is take an *empty* example file and proceed to create a chronlogy index. After completion, there exists a Database/Temp directory, with for every page in your database a file ending on ...txt.num. Now collect from those files as many lines as you need, and categorize them according to your system...

List of language-dependent files relative to the Programs directory (only the NL-variant is given):

5 Detailed description of the programs: indexing

This section contains detailed information on the Open Boek internals: scripts, logfiles and other stuff that you need when you want to develop your own Open Boek modules.

From the programdir two php programs can be called: admin.php and engine.php. They provide respectively for the indexing of the documents and the retrieval. In turn they can call includes and scripts in the directories Admin and Engine respectively. In this chapter we describe the stuff in Admin.

5.1 Prepare_data

The first program that will be run for a new database is *prepare_data.php*. Prepare_data keeps a log of its actions in the database directory as *prepare_data.log*. It calls in its turn prepare_pdf.php, prepare_html.php or prepare_txt.php to handle the three formats that Openboek accepts. Furthermore *wintok*, to parse the files and *checklang* to check the language.

- (if called with option pdf) it calls *pdftohtml* to extract from the pdf-file the individual pages as numbered HTML-files and separate images. It also creates an index-file, called name_ind.html, and OB uses this index file to keep track of the pages.
- it extracts the text proper from the HTML-files, so that SMART can later index those files, adding the markers <DOC page numpages docname>, <PAGE page numpages docname>, <TEXT> and <STOP> for the SMART preprocessor. The text is stored in Database/Data/docname.ob.txt
- it extracts the tokens from the individual pages (HTML-files) and stores them in *_token files. Dito for the HTML-tags, which are stored in the *_taglijst files and punctuation information (*_interpunction).

./Admin/eval_cardinals_NL.awk	translating cardinals and ordinals
./Admin/init_NL.awk	general rules for dutch texts
./Admin/templates.NL	
./Tagger/NL	directory for the POS-tagger
./Geocoding/Database/NL1_compleet.txt	
./General/chron_examples.ann.NL	MBL table for chronology
./General/loc_examples.ann.NL	MBL table for geography
./General/plaatsen.NL	provisional list
./General/countries.NL	nouns and adjectives that indicate countries
./General/stopw.NL	stopwords for indexing
./General/places.NL	alternative names for places
./General/help.NL.html	helpfile
./General/dialogs.NL	dialogs in the interface

Table 9: The language-dependent files

Field	Type	Null	Key	Default	Extra
D_key	int(11)	NO	PRI	NULL	auto_increment
cumpag	int(11)	YES	MUL	NULL	
numpag	int(11)	YES		NULL	
naam	varchar(64)	YES		NULL	
suffix	varchar(16)	YES		NULL	
indexed_chron	char(1)	YES		NULL	
indexed_loc	char(1)	YES		NULL	
indexed_keyw	char(1)	YES		NULL	
pos_tagged	char(1)	YES		NULL	
linked	char(1)	YES		NULL	
language	char(2)	YES		NULL	

Table 10: The table filelijst.

- finally it writes the files 'doc_loc' with the filenames (needed by SMART) and the table filelijst (see table 10) with a concordance of pagenumbers and documents to the home directory.

5.2 Creating the keyword indexes

At this point the keyword indexes can be created, after which Open Boek can already be used as an advanced VSM-based retrieval system. In the home directory, you will see a number of files, beginning with 'spec.'. These files govern the behaviour of SMART. It should not be necessary to change anything in those files, but note that if you want to use a list of stopwords, they should be called 'stopw.NL' (NL is the language, for every language a different one). These files have to be present, but they can be empty. We will assume that the binary 'smart' is copied to '/usr/local/bin'.

index_smart. This script calls SMART to create the frequency- and atc (tf.idf) indexes. The results are stored in the directories 'Docs' and 'Pages' respectively. Then the script *smprint* is called to create human- readable indices (word_weights.atc and word_weights.nnn). Finally it creates the 'inverted_file' files in Docs and Pages and loads them into the sql-table page_index.

Logs are kept in the database directory as *index_smart.log* and *index_time.log.*

5.3 The time indexes

classify_time. This script handles the recognition and indexing of chronology and other numeric data. It calls *wintok* and *numpick* to make lists of numbers in context. The script *paai_tag_time* recognizes whether the ex-

Field	Type	Null	Key	Default	Extra
page	int(11)	YES		NULL	
freq	int(11)	YES		NULL	
weight	float	YES		NULL	
word	varchar(25)	YES	MUL	NULL	

Table 11: Test_3, table page_index. Contains a regular index.

T_key	int(11)	NO	PRI	NULL	auto_increment
file	varchar(256)	YES		NULL	
bladzijde	int(11)	YES		NULL	
starttijd	bigint(20)	YES	MUL	NULL	
stoptijd	bigint(20)	YES	MUL	NULL	

Table 12: The time index.

pressions are chronological or spatial coordinates and creates the *_taglijst_chron files with the timespan tags for every page. Then, *index_time* extracts the 'Timespan' information from those files and stores it in 'tijdlijst' as an index (see table **??**).

The logfiles are : *classify_time.log*, *numpick.log*, *paai_tag_time.log* and *wintok.log.*

5.4 The location indexes

As already indicated, before indexing geographical names, you need first to POS tag the files. This is performed by *classify_pos.php*, which uses the Mbt tagger and the files in the Tagger directory. The POS tags are added as stand-off files in the data directories.

classify_loc.php. This script handles the recognition and indexing of place names from the list 'plaatsen_coordinaten.txt'. It calls *wintok mogelijke_plaats.php*, *kies_loc.php*, *class_loc_next_pag.php* and *loc_pick* to make lists of place names in context. The script *paai_tag_loc* recognizes whether the expressions are proper place names and creates the *_taglijst_loc files with the timespan tags for every page. Then, *index_loc* extracts the information from those files and stores it in 'loclijst' as an index.

The logfiles are : *classify_loc.log*, *loc_pick.log*, *paai_tag_loc.log* and *wintok.log.* There is also a *lit.log* that records which pages were not indexed because they were flagged as 'literature'.

5.5 Retrieval

Retrieval is based on the indexes created by the programs in the former section. The results are written to temporary sql-files prefixed with 'tmp...'

(see table 14. Every query has an unique number, so that the tmp-files can be inspected in case something unexpected happens. Old tmp-tables can be deleted from the administration menu.

- a php-script (index.php) is called in a browser. Keywords, chronological queries and geographical queries (class queries) are entered in separate fields. The intermediate results are stored in tmp-files, which then are joined.
- a less verbose interface is to be found in *display_query_balk.php.*
- after formulating the queries they are are parsed an checked for shortcuts in *parse_queries.php.*
  - the script *query_smart* calls smart with a query; generally as a backend of the php-interface script. It also can read the inverted files and perform a boolean query. 'Database.lst' is used to find the name of the document from the page. The SMART engine is used by creating a file with the commands that would be given from the interactive interface of SMART, and collecting the output from SMART in a file. Long live the Unix pipe! The logfile for this action is *query_smart.log*; the resultfile something like 'tmp_result_12345_key'.
  - *query_time* queries the 'tijdlijst' file. It also does a last check on consistency. logfile: *query_time.log.* The resultfiles look like 'tmp_12345_chron'and 'tmp_12345_chron_tmptijd. This last file is created to create a graph with 'tijdsgraaf'.
  - *query_loc* queries the 'loclijst' file. logfile: *query_loc.log.* The resultfile looks like 'tmp_12345_loc'.
  - KWIC queries are handled in *index_kwic.php*
- The final results are written to temporary files (see table **??**). It contains from left to right the weight, the absolute pagenumber, the page in the document and the document path. If both *query_time* and *query_smart* were called, the result is the join of both results. The resultfile looks like 'tmp_result_12345'.
- The php-interface and supporting scripts read this file and displays the list of pages and documents. These scripts reside in directory Page_display in the directory Engine in Programs.
- These script display the corresponding page, using the *combine* scripts *highlight* to highlight selected markups and where possible, to improve rendering. It leaves the following logfiles in the document-directory: tmp.html, combine.log, highlight.log, index_time.log and wintok.log.
- if a Google-like display is preferred (...

The queries are solved as follows: the temporal, geographical and keyword indexes are scanned for matches; the matches are stored in 'tmp_result_12345_key',

’tmp_12345_loc’ and ’tmp_12345_chron’. These files then are combined according to the genre of the query (boolean, frequency or advanced) and stored in the ultimate resultfile ’tmp_result_12345’.

```
tmp_chron_1200991778
tmp_chron_1200991778_tmptijd
tmp_result0_1200991778
tmp_result0_1200991778_docs
tmp_result0_1200991778_key
tmp_result1_1200991778
tmp_result2_1200991778
```

5.6 The server

The server, that enables users to upload pdf-files and to collect the chronological indices is *server.php*. It uses the package *Uber Uploader*[8], which should be installed in the *Server* directory. It is already in the Open Boek tarball. Please note that a download directory, Upload_dir, is already hardwired in the program.

[8] http://uber-uploader.sourceforge.net/

6 Database structure

This section covers the structure of the databases used by Open Boek. As of now (2008) the system consists of three tables (see table 15): “db_lijst”(see table 16), “geonames”(see table 17) and “bibliografie”(see table 18).

As an example, we will now describe the example-database Test_3:

7 The annotator

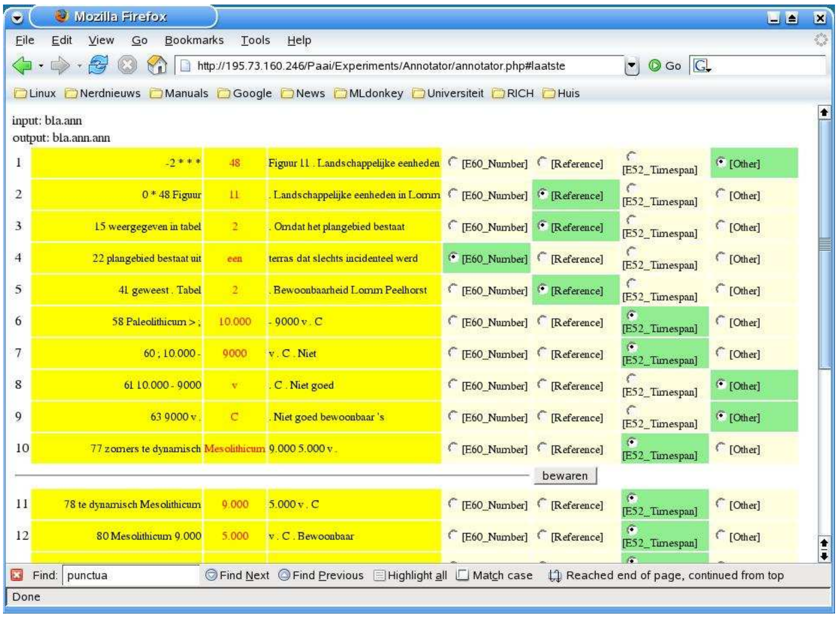
As Open Boek for its special functions depends on the existence of annotated examples, we have also added a simple web based annotation tool. It is called directly from your browser or from the Open Boek administrator interface.

To use the annotator, you must prepare a file with text windows (sequences of a certain number of words) with a focus of the feature that you want to classify and a label field for the assigned class. See 4.6 for an easy trick to create such files from existing pdf-documents.

As an example, consider the file “time_examples.ann” (table 26)

The file has nine features. The feature to be classified is in the column ’focus’ and is in our case a numeric, a cardinal or an ordinal. The purpose of annotation is to enter the correct label in the last column.

You can start the annotator by loading the URL http://.../annotator.php. Our annotator expects the file to be annotated to have the suffix “.ann”, and to have spaces as separators between the attributes. This file should be stored below the Programs directory and have the name ’Annotate’. When you start working with the annotator, new files also get a number in the filename, that is incremented after every save. This ensures that you have a complete history of your efforts, in case something bad happens.



http://195.73.160.246/Paai/Experiments/Annotator/annotator.php#laatste

input: bla.ann
output: bla.ann.ann

1	-2 * * *	48	Figuur 11 . Landschappelijke eenheden	○ [E60_Number]	○ [Reference]	○ [E52_Timespan]	● [Other]
2	0 * 48 Figuur	11	. Landschappelijke eenheden in Lomm	○ [E60_Number]	● [Reference]	○ [E52_Timespan]	○ [Other]
3	15 weergegeven in tabel	2	. Omdat het plangebied bestaat	○ [E60_Number]	● [Reference]	○ [E52_Timespan]	○ [Other]
4	22 plangebied bestaat uit	een	terras dat slechts incidenteel werd	● [E60_Number]	○ [Reference]	○ [E52_Timespan]	○ [Other]
5	41 geweest . Tabel	2	. Bewoonbaarheid Lomm Peelhorst	○ [E60_Number]	● [Reference]	○ [E52_Timespan]	○ [Other]
6	58 Paleolithicum > ;	10.000	- 9000 v . C	○ [E60_Number]	○ [Reference]	● [E52_Timespan]	○ [Other]
7	60 ; 10.000 -	9000	v . C . Niet	○ [E60_Number]	○ [Reference]	● [E52_Timespan]	○ [Other]
8	61 10.000 - 9000	v	. C . Niet goed	○ [E60_Number]	○ [Reference]	○ [E52_Timespan]	● [Other]
9	63 9000 v .	C	. Niet goed bewoonbaar 's	○ [E60_Number]	○ [Reference]	○ [E52_Timespan]	● [Other]
10	77 zomers te dynamisch	Mesolithicum	9.000 5.000 v .	○ [E60_Number]	○ [Reference]	● [E52_Timespan]	○ [Other]

bewaren

11	78 te dynamisch Mesolithicum	9.000	5.000 v . C	○ [E60_Number]	○ [Reference]	● [E52_Timespan]	○ [Other]
12	80 Mesolithicum 9.000	5.000	v . C . Bewoonbaar	○ [E60_Number]	○ [Reference]	● [E52_Timespan]	○ [Other]

Find: punctua

Reached end of page, continued from top

Done

Figure 17: Annotator for time_examples.ann.

The first time you select a file for annotation, you must enter the number of classes that you will be using, and press the button ’reload’. Then, indicate the number of features in the file, the focus field and the field with the class, but the annotator will already have computed them. After the first run, the annotator will save the values you have selected in a file with a .rc suffix and reload them automatically.

The annotation is straightforward: every line presents the classes you may to assign; just click on the corresponding radio button (see figure 17). When you are tired, press one of the buttons with “Save” that occur every ten lines; your work will be saved with the next highest number.

The structure of the annotation files is written in a rc-file that has the corresponding name.

Note that the file created by the annotator needs to be translated by 'General/translate_examples.awk' before it can be used by the system. To do this, enter 'sh translate_examples.awk < input-file > output-file' in the commandline.

7.1 Adding evaluation information

It is easy to apply this annotator as an evaluation tool. Given a database filled with what Open Boek assumes are the correct instances for every case, you only have to add a new label field with classes like '[correct]' and '[false]' and proceed to use these labels as the new classification. You can obtain such files by collecting from the directory database/Temp all files ending on '.classified' (see also 4.6). After tagging the instances by these labels, it is relatively easy to compute the performance of Open Boek for the given documents.

Nota Bene: the annotation task often is much easier if you sort the records on the focus column or any other criterium that ranks them in sensible groups.

8 Acknowledgements

This work was supported by NWO/CATCH under grant 640.002.401.

No Microsoft software was used in research or production of this document.

T_key	int(11)	NO	PRI	NULL	auto_increment
file	varchar(256)	YES		NULL	
bladzijde	int(11)	YES		NULL	
location	varchar(20)	YES		NULL	
country	varchar(3)	YES		NULL	
latitude	float	YES		NULL	
longitude	float	YES		NULL	

Table 13: loclijst2

Field	Type	Null	Key	Default	Extra
D_key	int(11)	NO	PRI	NULL	auto_increment
gewicht	int(11)	YES		NULL	
startpag	int(11)	YES		NULL	
pag	int(11)	YES		NULL	
docnaam	varchar(256)	YES		NULL	

Table 14: Test_3, table tmp_result_1229330873_0. Temporary table containing results from a query.

Tables_in_Openboek
db_lijst
geonames
bibliografie

Table 15: The main database, Openboek.

Field	Type	Null	Key	Default	Extra
L_key	int(11)	NO	PRI	NULL	auto_increment
naam	varchar(256)	YES		NULL	
aant_docs	int(11)	YES		NULL	
aant_pags	int(11)	YES		NULL	
root	char(255)	YES		NULL	

Table 16: Openboek, table db_lijst. Contains a list of all document databases and their attributes.

Field	Type	Null	Key	Default	Extra
geonameid	int(11)	NO	PRI	NULL	auto_increment
name	varchar(200)	YES	MUL	NULL	
asciiname	varchar(200)	YES		NULL	
alternatenames	varchar(4000)	YES		NULL	
latitude	float	YES	MUL	NULL	
longitude	float	YES	MUL	NULL	
feature_class	varchar(1)	YES		NULL	
feature	varchar(10)	YES		NULL	
country	varchar(2)	YES		NULL	
cc2	varchar(60)	YES		NULL	
admin1	varchar(20)	YES		NULL	
admin2	varchar(80)	YES		NULL	
admin3	varchar(20)	YES		NULL	
admin4	varchar(20)	YES		NULL	
population	int(11)	YES		NULL	
elevation	int(11)	YES		NULL	
gtopo30	int(11)	YES		NULL	
timezone	varchar(50)	YES		NULL	
modification	date	YES		NULL	

Table 17: Openboek, table geonames. Contains information for recognizing and plotting geographical features.

Field	Type	Null	Key	Default	Extra
pid	int(10)	NO	PRI	NULL	auto_increment
Issue	varchar(20)	YES		NULL	
Author	varchar(200)	YES		NULL	
Title	varchar(200)	YES		NULL	
Page	int(10)	YES		NULL	
File	varchar(200)	YES		NULL	
Date_created	int(10)	YES		NULL	
ISSN	varchar(50)	YES		NULL	
ISBN	varchar(50)	YES		NULL	
Link	varchar(100)	YES		NULL	

Table 18: Openboek, table bibliografie. Contains the metadata from the databases.

Tables_in_Test_3
chronlijst
filelijst
page_index
tmp_result_1229330873_0

Table 19: Example database: Test_3

Field	Type	Null	Key	Default	Extra
T_key	int(11)	NO	PRI	NULL	auto_increment
file	varchar(256)	YES		NULL	
bladzijde	int(11)	YES		NULL	
starttijd	bigint(20)	YES	MUL	NULL	
stoptijd	bigint(20)	YES	MUL	NULL	

Table 20: Test_3, table chronlijst. Contains the chronological index.

Field	Type	Null	Key	Default	Extra
D_key	int(11)	NO	PRI	NULL	auto_increment
cumpag	int(11)	YES	MUL	NULL	
numpag	int(11)	YES		NULL	
naam	varchar(64)	YES		NULL	
suffix	varchar(16)	YES		NULL	
indexed_chron	char(1)	YES		NULL	
indexed_loc	char(1)	YES		NULL	
indexed_keyw	char(1)	YES		NULL	
pos_tagged	char(1)	YES		NULL	
linked	char(1)	YES		NULL	
language	char(2)	YES		NULL	

Table 21: Test_3, table filelijst. Contains a list of files and their attributes.

Field	Type	Null	Key	Default	Extra
page	int(11)	YES		NULL	
freq	int(11)	YES		NULL	
weight	float	YES		NULL	
word	varchar(25)	YES	MUL	NULL	

Table 22: Test_3, table page_index. Contains a regular index.

T_key	int(11)	NO	PRI	NULL	auto_increment
file	varchar(256)	YES		NULL	
bladzijde	int(11)	YES		NULL	
starttijd	bigint(20)	YES	MUL	NULL	
stoptijd	bigint(20)	YES	MUL	NULL	

Table 23: Test_3, table chronlijst

T_key	int(11)	NO	PRI	NULL	auto_increment
file	varchar(256)	YES		NULL	
bladzijde	int(11)	YES		NULL	
location	varchar(20)	YES		NULL	
country	varchar(3)	YES		NULL	
latitude	float	YES		NULL	
longitude	float	YES		NULL	

Table 24: loclijst

Field	Type	Null	Key	Default	Extra
D_key	int(11)	NO	PRI	NULL	auto_increment
gewicht	int(11)	YES		NULL	
startpag	int(11)	YES		NULL	
pag	int(11)	YES		NULL	
docnaam	varchar(256)	YES		NULL	

Table 25: Test_3, table tmp_result_1229330873_0. Temporary table containing results from a query.

				focus					Label
telefoon	:	020	-	463	4848	Zeedijk	54	telefax	[Other]
AAI	’s	:	tussenbalans	1	januari	2000	,	Maastricht	[E52_Timespan]
o	25	-	30m	10	-	15m	3	03	[Other]
veen	0	0	0	0	1	411	20	veen	[Other]
de	hand	.	Figuur	17	(	links	)	coupe	[Reference]
Drie	fibulae	uit	de	eerste	helft	van	de	eerste	[E52_Timespan]

Table 26: The contents of time_examples.ann

Index

References

[1] Walter Daelemans, Jakub Zavrel, Ko van der Sloot, and Antal van den Bosch. Timbl: Tilburg memory based learner, version 5.1, reference guide. ilk technical report 04-02. Technical report, Tilburg University, 2004.

[2] H. Paijmans. Extraction of numeric data from multilingual archeological papers. In M. Ioannides, A. Addiso, and A. Georgopoulos, editors, *Digital heritage: proceedings of the 14th international conference on Virtual systems and Multimedia.* Archaeolingua, Budapest, 20-25 October 2008.

[3] J. J. Paijmans. Indexing texts with smart. *Linux Journal*, (36):24–26, april 1997.

[4] J.J. Paijmans and S. Wubben. Memory based learning and the interpretation of numbers in archaeological reports. In M-F Moens, T. Tuytelaars, and A.P. de Vries, editors, *Proceedings of the 7th Dutch-Belgian Information Retrieval Workshop*, pages 51–56, 2007.

[5] J.J. Paijmans and S. Wubben. Preparing archeological reports for intelligent retrieval. In Posluschny, K. Lambers, and I. Herzog, editors, *Proceedings of CAA-2007 (in press). Berlijn, Germany*, volume 10 of *Kolloquien zur Vor- und Frhgeschichte.* Dr. Rudolf Habelt GmbH, Bonn, 2007.

[6] G. Salton, editor. *The SMART retrieval system; experiments in automatic document processing.* Prentice-Hall, Englewood Cliffs, N. J. , 556 pp., 1971.

Files in the OB distribution (version 3.0)

```
./Coords
./Coords/style.css
./Coords/getcoords
./Coords/bt.css
./Coords/bt.gif
./Coords/rd2wgs
./Coords/BubbleTooltips.js
./Coords/coords2.php
./Engine
./Engine/combine
./Engine/query_smart.php
./Engine/highlight.org
./Engine/display_test.php
./Engine/chron_graph.php
./Engine/display_results.php
./Engine/parse_queries.php
./Engine/index_kwic.php
./Engine/chron_query.php
./Engine/display_questions.php
./Engine/Page_display
./Engine/Page_display/parms.php
./Engine/Page_display/combine
./Engine/Page_display/page_display.php
./Engine/Page_display/style.css
./Engine/Page_display/pasop.php
./Engine/Page_display/custombuttons.js
./Engine/Page_display/blind_index.php
./Engine/Page_display/lijst.php
./Engine/Page_display/highlight
./Engine/Page_display/greppo
./Engine/Page_display/pdf-page.php
./Engine/Page_display/wz_tooltip.js
./Engine/Page_display/index.php
./Engine/Page_display/bla-1.html
./Engine/Page_display/tagcheck
./Engine/Page_display/maak_page.php
./Engine/Page_display/hitknoppen.php
./Engine/Page_display/combine_org
./Engine/Page_display/hitlijst.php
./Engine/Page_display/leeshits.php
./Engine/Page_display/knoppen.php
./Engine/display_query_balk.php
./Engine/loc_query.php
```

```
./Engine/combine.org
./Engine/tijdsgraaf
./Engine/place_graph.php
./radio.gif
./radio.png
./Admin
./Admin/metadata_complex.php
./Admin/adresser.php
./Admin/class_loc_next_pag.php
./Admin/evaluate.awk
./Admin/templates.raw
./Admin/create_db.php
./Admin/flash_upload.php
./Admin/prepare_txt.php
./Admin/eval_cardinals_DE.awk
./Admin/eval_cardinals_EN.awk
./Admin/ob_manual_3.pdf
./Admin/prepare_pdf.php
./Admin/voortgang.php
./Admin/eval_cardinals_NL.awk
./Admin/chron_pick
./Admin/classify_chron.php
./Admin/protocol_prepdata
./Admin/mogelijke_plaats.php
./Admin/init_language.awk
./Admin/match.php
./Admin/database.result
./Admin/prepare_data.php
./Admin/eval_cardinals.awk
./Admin/init_DE.awk
./Admin/init_EN.awk
./Admin/eval_roman.awk
./Admin/prepare_html.php
./Admin/tag_masks
./Admin/protocol_loc
./Admin/ttt_test
./Admin/landnamen.php
./Admin/database.tenfold
./Admin/openboek.rc
./Admin/classify_pos.php
./Admin/index_loc
./Admin/kies_loc.php
./Admin/init_NL.awk
./Admin/index_chron
./Admin/index_smart
```

```
./Admin/upload.php
./Admin/tellen
./Admin/classify_boiler_1.php
./Admin/check_lit
./Admin/checklang
./Admin/wintok
./Admin/templates.NL
./Admin/splittext.pl
./Admin/texput.log
./Admin/posprepare.pl
./Admin/query_int_admin.php
./Admin/10-fold
./Admin/metadata.php
./Admin/paai_tag_loc
./Admin/template_maker
./Admin/template_maker.awk
./Admin/classify_loc.php
./Admin/import_db.php
./Admin/paai_tag_chron
./Icons
./Icons/arrowdubb_left.gif
./Icons/ob_logo.jpg
./Icons/arrowdubb_right.gif
./Icons/style.css
./Icons/arrow_right.gif
./Icons/openboek.jpg
./Icons/Icons.txt
./Icons/lijst.php
./Icons/arrowleft.png
./Icons/arrow_left.gif
./Icons/rich.jpg
./Icons/arrowrightdub.gif
./Icons/arrowrightdub.png
./Icons/index.php
./Icons/arrowleftdub.gif
./Icons/arrowleftdub.png
./Icons/arrowright.png
./Icons/knoppen.php
./openboek.rc.test
./smart
./Needed
./Needed/smart
./Needed/timbl-6.1.5.tar.gz
./Needed/mbt-3.1.3.tar.gz
./sorry.php
```

```
./Server
./Server/effe
./Server/effe/Uber-Uploader_6.3.5
./Server/effe/Uber-Uploader_6.3.5/html
./Server/effe/Uber-Uploader_6.3.5/html/images
./Server/effe/Uber-Uploader_6.3.5/cgi-bin
./Server/ubr_lib.php
./Server/CREDITS.TXT
./Server/ubr_get_progress.php
./Server/ubr_finished.php
./Server/ubr_finished_lib.php
./Server/ubr_upload.pl
./Server/config
./Server/ubr_file_upload.php
./Server/ubr_file_upload.js
./Server/ubr_set_progress.php
./Server/CHANGE_LOG.TXT
./Server/images
./Server/images/progress_bar_white.gif
./Server/images/progress_bar_blue.gif
./Server/ubr_link_upload.php
./Server/INSTALL_AND_FAQ.TXT
./Server/ubr_ini.php
./Server/ubr_image_lib.php
./Server/ubr_default_config.php
./Tagger
./Tagger/DE
./Tagger/DE/bla
./Tagger/DE/test
./Tagger/DE/train.lex
./Tagger/DE/train.5paxes
./Tagger/DE/conversietabel_tags_DE.txt~
./Tagger/DE/train.unknown.dFapsss
./Tagger/DE/train
./Tagger/DE/conversietabel_tags_DE.txt
./Tagger/DE/train.settings~
./Tagger/DE/train.lex.ambi.05
./Tagger/DE/train.settings
./Tagger/DE/train.known.ddfa
./Tagger/DE/train.top100
./Tagger/EN
./Tagger/EN/dat
./Tagger/EN/test
./Tagger/EN/train.lex
./Tagger/EN/train.5paxes
```

```
./Tagger/EN/train.unknown.dFapsss
./Tagger/EN/train
./Tagger/EN/conversietabel_tags_EN.txt
./Tagger/EN/train.settings~
./Tagger/EN/train.lex.ambi.05
./Tagger/EN/train.settings
./Tagger/EN/train.known.ddfa
./Tagger/EN/train.top100
./Tagger/NL
./Tagger/NL/bla
./Tagger/NL/train.lex
./Tagger/NL/train.5paxes
./Tagger/NL/train.unknown.dFapsss
./Tagger/NL/train
./Tagger/NL/train.settings~
./Tagger/NL/train.lex.ambi.05
./Tagger/NL/train.unknown.psssdwFaw
./Tagger/NL/train.settings
./Tagger/NL/train.known.dwdwfWaw
./Tagger/NL/NL_converted.tgz
./Tagger/NL/NL_converted
./Tagger/NL/test.posready.txt
./Tagger/NL/train.known.ddfa
./Tagger/NL/train.unknown.psssdwFaw.bck
./Tagger/NL/train.known.ddfa.wgt
./Tagger/NL/train.known.dwdwfWaw.bck
./Tagger/NL/train.known.dwdwfWaw.wgt
./Tagger/NL/train.top100
./Tagger/NL/train.ambi.05
./Tagger/NL.tgz
./custombuttons.js
./logfile
./admin.php
./format
./gnucom
./Upload_dir
./openboek.rc
./server.php
./index.php
./engine.php
./annotator.css
./paaitest.php
./timbl-6.1.5.tar.gz
./protocol
./General
```

```
./General/chron_examples.ann.DE
./General/chron_examples.ann.EN
./General/chron_examples.ann.NL
./General/engels.chron
./General/eras.rc.DE
./General/eras.rc.EN
./General/eras.rc.NL
./General/translate_examples.awk
./General/spec.pro
./General/functions.php
./General/chronologie nederland.txt
./General/display_databases.php
./General/spec.expcoll
./General/smprint
./General/common_words
./General/rc_lezer.php
./General/complextypen
./General/voorbeeld_DU
./General/spec.Docs.atc
./General/dialogs.EN
./General/dialogs.NL
./General/spec.Pages
./General/loc_examples.ann.DE
./General/loc_examples.ann.EN
./General/loc_examples.ann.NL
./General/plaatsen_coords
./General/help.EN.html
./General/#rc_lezer.php#
./General/annotator.php
./General/spec.Docs
./General/coords_ned
./General/chron_examples.DE.VBD.1.ann
./General/plaatsen.NL
./General/eras.all.rc
./General/countries.EN
./General/countries.NL
./General/Annotate
./General/Annotate/bla
./General/Annotate/hop
./General/Annotate/eval
./General/Annotate/time_examples_dutch.rc
./General/Annotate/rdmz_paai_tag_results.rc
./General/Annotate/eval2
./General/Annotate/timespans_test.ann
./General/Annotate/chron_examples.ann.NL.org.1.ann
```

```
./General/Annotate/york.rc
./General/Annotate/loc_examples.rc
./General/Annotate/chron_examples.DE.VBD.ann
./General/Annotate/chron_examples.ann.EN.1.ann
./General/Annotate/chron_examples.ann.EN.2.ann
./General/Annotate/chron_examples.ann.EN.3.ann
./General/Annotate/loc_examples.ann.result
./General/Annotate/chron_examples.ann.NL.org.rc
./General/Annotate/york_2.rc
./General/Annotate/loc_examples.ann.NL
./General/Annotate/chron_examples.DE.VBD.rc
./General/Annotate/loc_examples.ann.tenfold
./General/Annotate/time_examples_dutch.ann
./General/Annotate/rdmz_class_eval.ann
./General/Annotate/chron_examples.DE.VBD.1.ann
./General/Annotate/chron_examples.DE.VBD.2.ann
./General/Annotate/chron_examples.ann.EN.rc
./General/Annotate/york.ann
./General/Annotate/loc_examples.1.ann
./General/Annotate/loc_examples.2.ann
./General/Annotate/10-fold
./General/Annotate/chron_examples.ann.NL.org.ann
./General/Annotate/rdmz_class_eval.rc
./General/Annotate/rdmz_class_evaluated
./General/Annotate/loc_examples.ann
./General/Annotate/rdmz_paai_tag.results.evaluated
./General/Annotate/york_2.ann
./General/Annotate/help.php
./General/Annotate/rdmz_paai_tag_results.ann
./General/Annotate/chron_examples.ann.EN.ann
./General/help.NL.html
./General/stopw.EN
./General/stopw.NL
./General/spec.Pages.atc
./General/spec.default
./General/places.DE
./General/places.EN
./General/places.NL
./General/help.php
./test.html
./mbt-3.1.3.tar.gz

\pagebreak

\section*{List of database specific files and their use}
```

```
\begin{verbatim}
/Data/*             : Directory containing the data used in the specific
                      database, organised per document in folders.
/Pages/*            : Directory containing indexes used by Open Boek.
/Timeloc/*          : Directory containing chronological indexes.
/common_words       : Text file containing a number of common words wich
                      are not taken into account while indexing.
/doc_loc            : Text file containing the location of the text files
                      of the documents for indexing by the SMART engine.
/eras.rc*           : A number of text files containing chronologies.
/spec.*             : A number of text files used by the SMART engine.
/stopw*             : A number of text files containing a number of common
                      words wich are nog taken into account while indexing,
                      in different languages.
/[name database].rc : A text file containing general information about the
                      database, such as language preferences.
```

Files used for the MBL examples

```
01+inleidings.html                    02+doelstellingen+en+organisaties.html
03+Zandmaass.html                     04+Grensmaass.html
05+natte+archeologies.html            06+steentijds.html
07+metaaltijds.html                   08+romeinse+tijds.html
09+middeleeuwens.html                 10+conclusiess.html
11+samenvattings.html                 12+Zusammenfassungs.html
13+publicatiess.html                  HOP1\_Gasleidings.html
AAIrap14-1.html                       AAIrap14-10.html
AAIrap14-11.hml                       AAIrap14-12.html
AAIrap14-13.html                      AAIrap14-14.html
AAIrap14-15.html                      AAIrap14-4.html
AAIrap14-5.html                       AAIrap14-6.html
AAIrap14-7.html                       AAIrap14-8.html
AAIrap14-9.html                       AAIrap20s.html
AAOrap02s.html                        AAOrap14s.html
AAOrap15s.html                        AAOrap22s.html
AAOrap29s.html                        AAOrap33s.html
AAOrap36s.html                        AAOrap38s.html
Hanzelijn7s.html                      NO1328-LOEBs.html
NO1342-BOHAs.html                     NO1353-VREIs.html
RA1156-NEBENs.html                    RA969-NLDAs.html
RAM\_79\_01\_Hoge\_Vaart-A27s.html    RAM\_79\_03\_Hoge\_Vaart-A27s.html
RAM\_79\_04\_Hoge\_Vaart-A27s.html    RAM\_79\_05\_Hoge\_Vaart-A27s.html
Rapport+86s.html                      archol\_06s.html
archol\_08s.html                      archol\_15s.html
archol\_18s.html                      archol\_21s.html
archol\_22s.html                      archol\_26s.html
archol\_27s.html                      archol\_30s.html
archol\_37s.html                      archol\_41s.html
fratsen\_1s.html                      fratsen\_3s.html
fratsen\_5s.html                      inhoudsopgaves.html
ockenburgh-jaarverslag-1993s.html     page5.html
F1+format+selectieadvies+waarderend+onderzoek+Maaswerkens.html
F2+format+programma+van+eisen+waarderend+onderzoek+Maaswerkens.html
F4+format+standaard+bepalingen+veldwerk+Maaswerkens.html
F5+format+standaard+bepalingen+uitwerken+Maaswerkens.html
F8+format+standaard+bepalingen+eindrapport+Maaswerkens.html
14+medewerkers+Projectteam+Archeologie+Maaswerkens.html
RAP+515\_4100420\_Eelde+Kosterijwegs.html
RAP+521\_4100020\_Beesel+Hoeve+Oud+Waterloos.html
RAP+558\_4094100\_Ede+Tuinderslaans.html
Selectieadvies+definitief+onderzoek+Lomms.html
HIO01\_project\_metainformaties.html
```