Journal of Digital Imaging

Design and Implementation of an Open Source Indexing Solution for a Large Set of Radiological Reports and Images

T. Voet, P. Devolder, B. Pynoo, J. Vercruysse, and P. Duyck

This paper hopes to share the insights we experienced during designing, building, and running an indexing solution for a large set of radiological reports and images in a production environment for more than 3 years. Several technical challenges were encountered and solved in the course of this project. One hundred four million words in 1.8 million radiological reports from 1989 to the present were indexed and became instantaneously searchable in a user-friendly fashion; the median query duration is only 31 ms. Currently, our highly tuned index holds 332,088 unique words in four languages. The indexing system is feature-rich and language-independent and allows for making complex queries. For research and training purposes it certainly is a valuable and convenient addition to our radiology informatics toolbox. Extended use of open-source technology dramatically reduced both implementation time and cost. All software we developed related to the indexing project has been made available to the open-source community covered by an unrestricted Berkeley Software Distribution-style license.

KEY WORDS: Open source, radiology information systems (RIS), PACS integration, medical informatics applications, radiology reporting, document indexing, search engines

BACKGROUND

Whenever a person or an institution builds a large enough collection of anything, searching through it becomes tedious or even unfeasible. An obvious solution is indexing: the process of compiling a list of pointers to items in the collection based on alphabetical order or any other item attribute. Indexing can add useful metadata, which is data about data, to a collection. A well-known example is the Science Citation Index for papers published in academic journals as envisioned by Eugene Garfield in 1955.[1]

Hospitals typically generate a lot of data about patients, their examinations, and their treatments. Due to the patient-oriented workflow, hospital information systems have a strong patient-centric architecture; hence, problems tend to arise whenever information across patients is required. For purposes of business intelligence, data mining and data warehousing can provide an adequate solution. When cross-patient information is needed in a research or training context, collecting medical data will become labor-intensive or practically impossible if the relevant data only exist in a free textual form. A solution for this problem consists in building an index of all free-text words that allows for efficient queries.

In our institution, the university hospital closely associated with the largest Belgian university, close to two million radiological reports and corresponding images were made instantaneously searchable by implementing a language-independent, feature-rich indexing solution. This paper hopes to share the insights we experienced during designing, building, and running an indexing system in a production environment for more than 3 years. Extended use of

From the Department of Radiology, Ghent University Hospital and Ghent University, De Pintelaan 185, 9000, Ghent, Belgium.

Correspondence to: T. Voet, Radiology, Ghent University Hospital and Ghent University, De Pintelaan 185, 9000, Ghent, Belgium; tel: +32-9-2403910; fax: +32-9-2404421; e-mail: Tony.Voet@uzgent.be

Online publication 9 August 2007
doi: 10.1007/s10278-007-9055-2

open-source technology dramatically reduced both implementation time and cost.

METHODS

To achieve a complete indexing solution that is ready for a production environment, we designed a system consisting of five major sections: the report warehouse, the search engine, specific import filters, efficient indexing, and finally the user interface. These sections correspond to five discrete phases in the implementation scheme. An overview of the architecture is presented in Figure 1.

Report Warehouse

Indexing a large set of reports requires fast access to all text files to reduce the time needed to build such an index. Our radiology information system (RIS) did not comply with this rule because reports were stored in an Oracle database using a Lempel–Ziv related compression scheme.[2] To counter this problem we decided to set up an independent report warehouse that did not have an impact on RIS performance.

We started out by recycling two 5-year-old servers and fitting both of them with enough hard disk capacity (160 gigabytes) and a small uninterruptible power supply. We chose to run the open-source Free Berkeley Software Distribution (FreeBSD) operation system,[3] which had the important ability to maintain disk integrity after a crash or power outage. This technique, called soft updates, works by properly ordering file system metadata-writes to guarantee consistency and was introduced by Marshall Kirk McKusick.[4] Both servers were configured as a high-availability cluster.

We achieve synchronization with the RIS through two parallel processes. First, a structured query language (SQL) statement interrogates the RIS to find out which reports were cancelled since the last synchronization. Subsequently, these cancelled reports are removed from the report warehouse. The second process is similar to the first; a list of new reports and reports updated with an addendum is obtained. Next, by means of another SQL statement, the report binary large objects are retrieved from the RIS database and decompressed. Alternatively, we could have been using a standardized, but more complex, method of capturing outbound Health Level 7[5] observation results unsolicited messages originating from RIS. At the time of implementation this method had the major disadvantage of only supporting converted flat text representations of the reports; meanwhile, our RIS gained the possibility to send out rich text format (RTF), retaining report layout and style. In an ideal situation, indexing should be built right into RIS; this way both extracting and synchronizing reports and data become obsolete. Our report

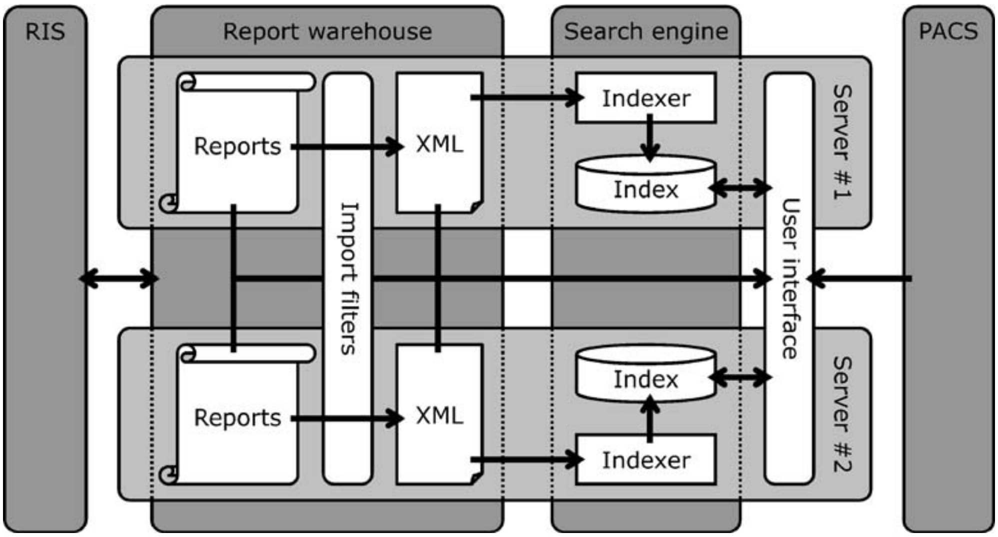


Fig 1. The architecture of the complete indexing system.

warehouse only contains validated reports by policy.

Search Engine

With the advent of the Internet and ever-increasing computing power, indexing software became widely available. Our report warehouse made it possible to plug in and evaluate several search engines with a limited set of reports. Due to our characteristic European setting, language independence was needed to support Dutch, French, English, and German reports. The ability to include RIS data in the index as report metadata was also an important factor; this allows searching for (or sorting on) specific authors and requesting physicians and departments and other administrative information. Four more features we required were phrase and wildcard searching, Boolean logic, and the possibility to exclude common stop words. The section on semantics in this paper contains a detailed motivation for why these last four features are essential in a medical context.

Both commercial and open-source indexing engines were tested and compared, but in-depth evaluation reports have not been included here because they have become seriously outdated after more than 3 years. Commercial engines were difficult or impossible to integrate due to their closed-source nature. Most engines could be discarded quickly because of a missing feature or a time-consuming indexing process. Note that the Google Search Appliance[6] was not available in Europe at that time. We stopped looking for other candidates after we evaluated the highly recommended Swish-e indexing engine,[7] which had all the desired features and a very efficient indexing process.

Import Filters

Our evaluation showed that the filters provided by most search engines, including Swish-e, had issues with importing our reports. Most problems were related to file formats; old formats were converted incorrectly, international characters often disappeared, and automatically generated text fragments (Microsoft Word AutoText) were not being filled in. One issue was observed across file formats: hyphenated words did not always make it into the index in one piece.

Solving these problems required programming import filters for all historically and currently used file formats: both American Standard Code for Information Interchange flat text and flat text containing multiple international character sets, WordPerfect, Microsoft Word, and RTF. These in-house developed filters properly undo hyphenation and combine the imported report with corresponding RIS data into a single Extensible Markup Language (XML) file ready for indexing, an example of which is presented in Figure 2. At our institution there is no mandatory structure for radiological reports. Additionally, we have put a quality assurance process in place; whenever word processing software or a report template is changed, the import results are tested on international characters, automatically generated fragments, and hyphenation.

Efficient Indexing

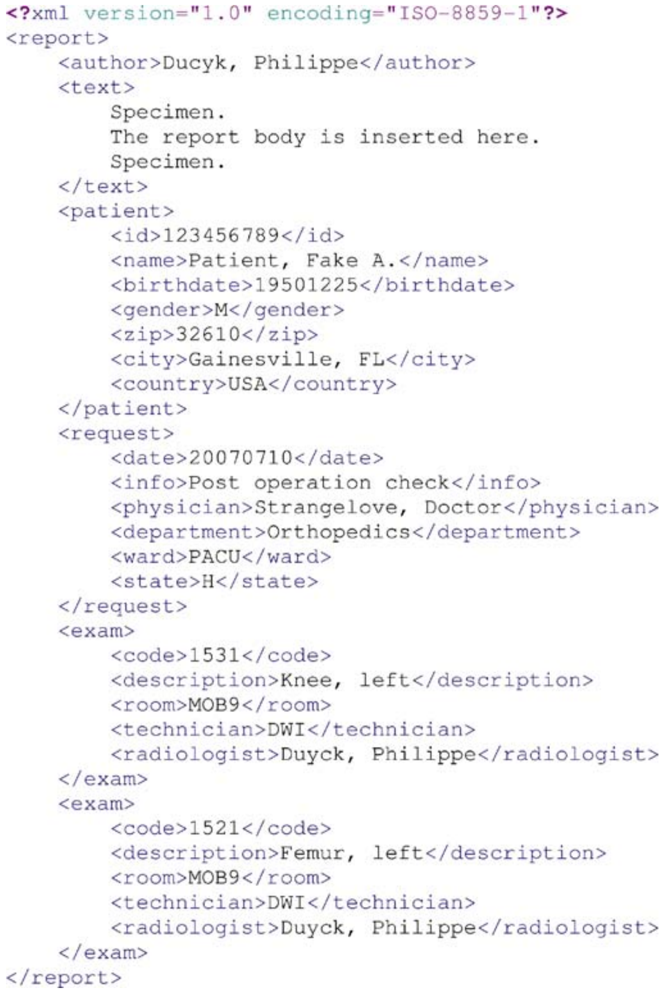
Indexing without tuning becomes very computing-intensive as the collection of document content, and thus the number of word/document combinations, grows. Fortunately, limiting the size of the index itself strongly decreases the computing power needed. In our situation, patient and study IDs in the radiological reports were safely excluded from the index without loss of information; using the metadata from the RIS yielded the same results and was more efficient.

A second technique consists in excluding some, but certainly not all, of the very frequent words present in almost all documents. These common stop words were found by building an index and evaluating the significance of the thousand most frequent words. Amongst the 39 manually excluded stop words, good examples like “a,” “the,” “of,” and closing statements like “greetings” were found. The very frequent word “no” was not excluded because its meaning was highly significant in combination with other keywords.

In the section on semantics we will show that excluding common stop words leads to semantic advantages. We will also discuss building indexes in an incremental fashion in the section on scalability.

User Interface

Due to our completely digital and highly optimized reporting workflow,[8] the only good

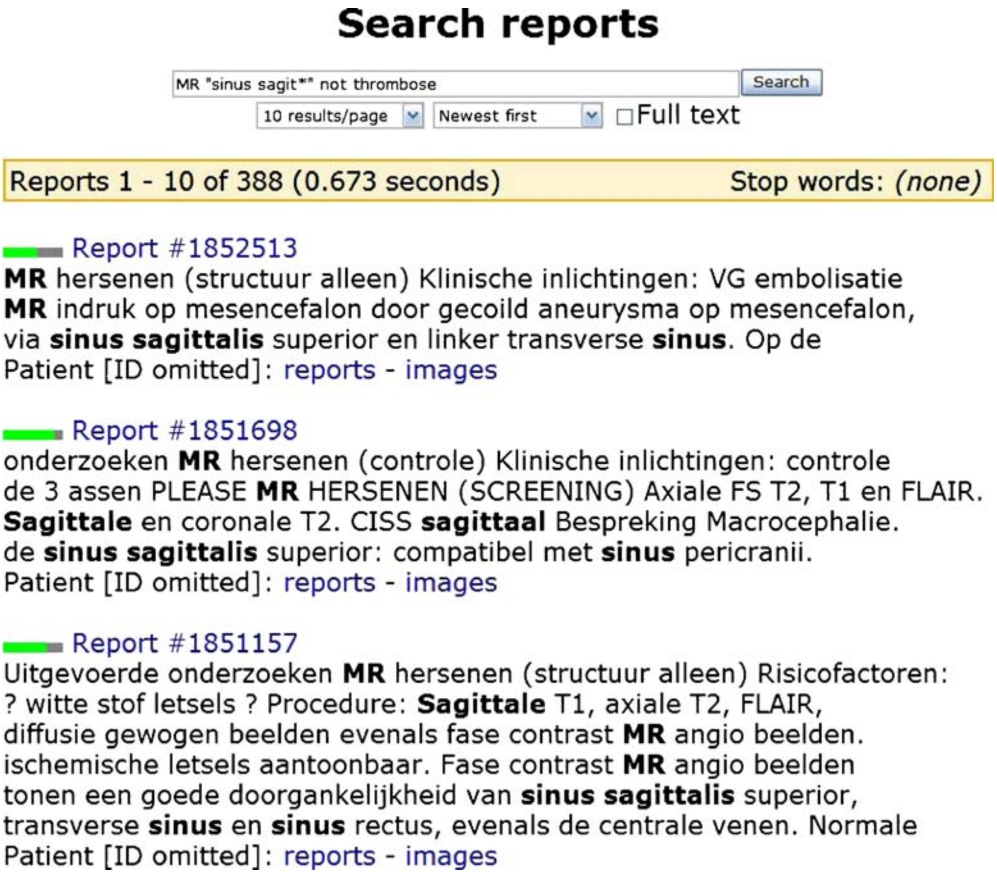


```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<report>
    <author>Ducyk, Philippe</author>
    <text>
        Specimen.
        The report body is inserted here.
        Specimen.
    </text>
    <patient>
        <id>123456789</id>
        <name>Patient, Fake A.</name>
        <birthdate>19501225</birthdate>
        <gender>M</gender>
        <zip>32610</zip>
        <city>Gainesville, FL</city>
        <country>USA</country>
    </patient>
    <request>
        <date>20070710</date>
        <info>Post operation check</info>
        <physician>Strangelove, Doctor</physician>
        <department>Orthopedics</department>
        <ward>PACU</ward>
        <state>H</state>
    </request>
    <exam>
        <code>1531</code>
        <description>Knee, left</description>
        <room>MOB9</room>
        <technician>DWI</technician>
        <radiologist>Duyck, Philippe</radiologist>
    </exam>
    <exam>
        <code>1521</code>
        <description>Femur, left</description>
        <room>MOB9</room>
        <technician>DWI</technician>
        <radiologist>Duyck, Philippe</radiologist>
    </exam>
</report>
```

Fig 2. Example of a XML file containing the imported report and its corresponding RIS metadata.

place to put the search interface was right at the radiologists' fingertips. To attain this, we created a full front-end integration with RIS and the picture archiving and communication system (PACS). By means of a generic search button, the interface can be launched from any RIS or PACS context. The integration also includes a security layer serving two distinct purposes. Firstly, our existing single-sign-on/single-sign-off mechanism allows radiologists and researchers to provide their authentication credentials using an encrypted hardware key, defying the need to type in many usernames and passwords. The second function of the security layer consists of auditing the search queries and the displayed reports.

Because Internet technology was being used, it was straightforward to make a web-based interface that resembles well-known Internet search engines. Such simple and user-friendly interfaces lower the threshold for new radiologists. As shown in Figure 3, users simply enter the desired keywords, possibly accompanied by parentheses and the binary operators "and," "or," and "not." Wildcards are denoted by an asterisk, and quotes can be used for phrase searching. Between keywords or phrases, an implicit "and" is assumed. The user can quickly pick a previous query from his complete query history. The results can be sorted on a calculated page rank (default), descending and ascending report time, and RIS metadata. Useful query feedback is shown: the number of results, the query duration, and the excluded stop words. Fast reviewing of reports is essential; to meet this end, the user can choose between displaying summa-



MR "sinus sagit*" not thrombose

10 results/page · Newest first · □Full text

Reports 1 - 10 of 388 (0.673 seconds) Stop words: *(none)*

Report #1852513
MR hersenen (structuur alleen) Klinische inlichtingen: VG embolisatie **MR** indruk op mesencefalon door gecoild aneurysma op mesencefalon, via **sinus sagittalis** superior en linker transverse **sinus**. Op de
Patient [ID omitted]: reports - images

Report #1851698
onderzoeken **MR** hersenen (controle) Klinische inlichtingen: controle de 3 assen PLEASE **MR** HERSENEN (SCREENING) Axiale FS T2, T1 en FLAIR. **Sagittale** en coronale T2. CISS **sagittaal** Bespreking Macrocephalie. de **sinus sagittalis** superior: compatibel met **sinus** pericranii.
Patient [ID omitted]: reports - images

Report #1851157
Uitgevoerde onderzoeken **MR** hersenen (structuur alleen) Risicofactoren: ? witte stof letsels ? Procedure: **Sagittale** T1, axiale T2, FLAIR, diffusie gewogen beelden evenals fase contrast **MR** angio beelden. ischemische letsels aantoonbaar. Fase contrast **MR** angio beelden tonen een goede doorgankelijkheid van **sinus sagittalis** superior, transverse **sinus** en **sinus** rectus, evenals de centrale venen. Normale
Patient [ID omitted]: reports - images

Fig 3. The user interface for radiologists and researchers.

rized or full reports—in both cases, keywords are highlighted and the matching ranking score is displayed graphically. With a single click of the mouse, radiologists can open both images in PACS and the historical reports of the selected patient.

Alas, trying to comprehend the ranking code of Swish-e is not for the faint of heart because of its inherent complexity and the speed optimizations used. The algorithms take into account the binary structure of the search query, details of keyword occurrences, and the relative frequency of words in the index. Luckily, unlike searching the Internet, queries in a radiological context tend to be very specific (see results), making fine-tuning of ranking unnecessary.

RESULTS

Preceding the go-live of our indexing solution, over one million reports and associated data were extracted from the RIS. Some filter fine-tuning was necessary to pass all reports through the layer of import filters. The initial import was spread over a 14-day period without performance penalties for production systems. At the time of writing, our report warehouse contained 1.8 million reports and corresponding administrative data, their XML versions and backup copies not included. The system holds 7.2 million files in total. The oldest report dates back to 1989, but not many reports of that era were available in digital form.

Each night the index is rebuilt completely, which facilitates early detection of disc decay because all XML reports get reread in the course of this process. Alternatively, incremental reindexing could have been used. Building a full index is very efficient; it only takes 76 min worth of CPU time on 8-year-old hardware to process 104 million words. Currently, the index contains 332,088 unique words in four languages, including names of patients and requesting physicians. Alas,

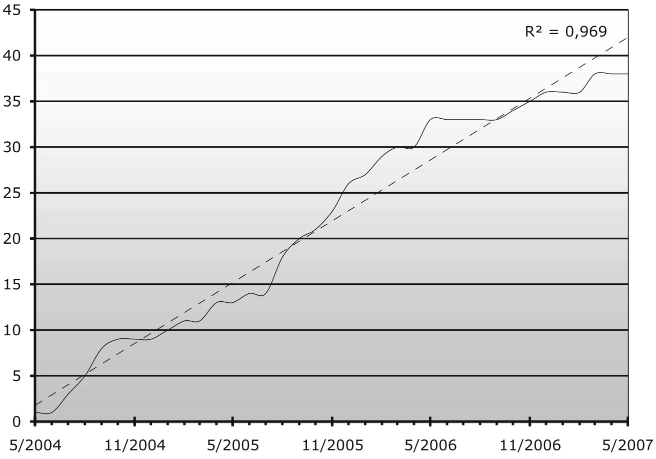


Fig 4. The increase in users over time.

all spelling mistakes ever made by our typists and radiologists are also included.

The usefulness of the indexing solution is indicated by the fact that all resident radiologists make use of it despite the nonmandatory character. Figure 4 illustrates the nearly linear increase in users since the initial deployment in May 2004.[9]

Indexing reports added useful metadata to our collection of radiological examinations. Early generations of the RIS did not track the supervisor of a radiologist in training; instead, the supervisor's name was added manually to the text of a report. Indexing made it possible to retrieve historical supervisor information.

Having an independent report warehouse available that contains all radiological reports avoids vendor lock-in. During PACS implementation we encountered a 6-month delay in the PACS/RIS front-end integration due to a difficult multivendor situation. We successfully avoided this delay by making historical reports available at the radiologists' work stations through the report warehouse.

Query Analysis

The search engine was consulted 7,071 times by our radiologists mainly to find interesting cases for research or training purposes. Occasionally, a query is performed to find a recent clinical exam in a fast and convenient manner.

Table 1 presents the usage of search engine features. Apparently, 61% of the radiologists are aware of and use at least one of the features despite the lack of formal training; 39% of them have never used a search engine feature. Nine in ten queries consist of keywords only, which means that results are mostly found without having to resolve to more complicated queries. Users tend to start out with a simple query and quickly skim across the clearly presented and highlighted results. Binary operators turned out to be the most popular feature, especially the "and" operator, which is used in 4.6% of the queries to narrow the returned results. Phrase and wildcard searching each are used in 3% of the queries, whereas parentheses account for 0.6%.

Querying the index is very fast; the median query duration is 31 ms. The most complex query took 2.6 s: results for each keyword need to be binary combined and ranked; a processing delay is noticeable in the case of very frequent keywords or when broad wildcards are used. The users enter fairly specific queries; medially, only 36 results are returned. A statistical analysis of the query duration and the number of results can be found in Table 2. The cumulative frequency distribution

Table 1. The Usage of Search Engine Features

Feature	Queries ($n = 7{,}071$)	Users ($n = 38$)
Keywords only	6,344 (89.7%)	15 (39%)
Phrases	215 (3.0%)	14 (37%)
Wildcards	208 (2.9%)	14 (37%)
Binary operator "and"	325 (4.6%)	15 (39%)
Binary operator "not"	117 (1.7%)	9 (24%)
Binary operator "or"	83 (1.2%)	8 (21%)
Parentheses	40 (0.6%)	6 (16%)

Table 2. Statistical Analysis of the Query Duration and the Number of Results

	Query Duration (ms)	Number of Results
Median	31	36
Maximum	2,598	962,629
Mean	129	5,878
Standard deviation	274	59,286
Skewness	4.4	15
Kurtosis	24	237

of the query duration, presented in Figure 5, can give the reader an idea of the response rate of the search engine. Caching performed by the operating system has a strong influence on the query duration because 36% of the queries yield results within the 13-ms average access time of the disk drive used; Figure 6 visualizes the relevant detail of the frequency distribution.

Open Source

Our indexing solution is a prime example of an open-source success story; extended use of this technology dramatically reduced both implementation time and cost. Designing and building the system took a mere 6 weeks, which was less than the “red tape” time needed to approve a commercial offering. Apart from the implementation man-hours, the total cost of ownership consists of low-budget hardware and support costs. Support calls are rare due to the high-availability design of the system and the excellent stability of the Swish-e search engine. In fact, users almost never call; the interface is very user friendly and similar to well-known interfaces of Internet search engines. Furthermore, passwords cannot be forgotten because of the single-sign-on mechanism. About four times a year, manual intervention is required when the synchronization link with the RIS breaks down; in this case, the administrator is automatically notified by e-mail. There has only been one hardware failure during the system’s lifetime, a broken power supply, but this did not cause any downtime.

All software we developed in the course of this project has been made available[10] to the open-source community covered by an unrestricted BSD-style license. The source code included linking and synchronization with a RIS, five import filters, the security layer and audit logging, the complete user interface, and the integration code for both RIS and PACS.

DISCUSSION

Although indexing technology has undoubtedly been used in a clinical setting before, the authors believe they have followed a novel approach in designing and implementing a complete and scalable indexing solution. This high-availability and high-performance system has been running without issues for more than 3 years in a production environment. Furthermore, the independent and flexible design of the platform allows for hosting other applications; several research projects have already found a home under its wings. In this section, Internet indexing technology, scalability, and semantics are briefly discussed.

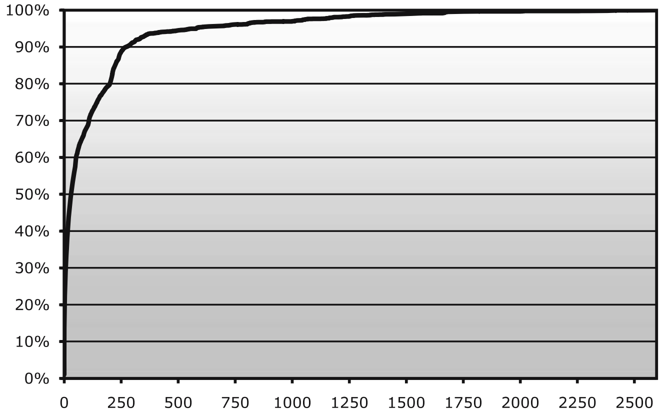


Fig 5. The cumulative frequency distribution of the query duration (ms).

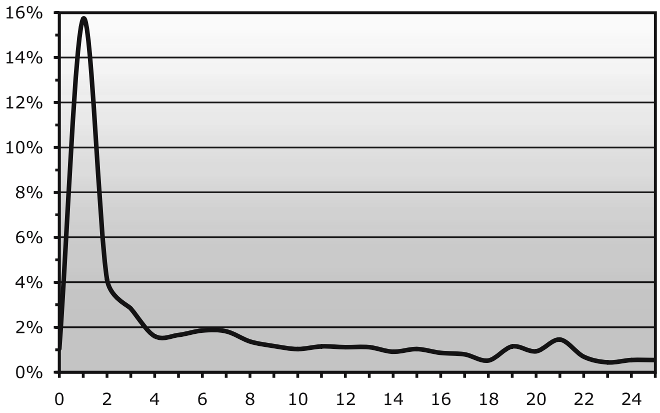


Fig 6. Detail of the frequency distribution of the query duration (ms), showing the influence of caching performed by the operating system.

Internet Indexing Technology

Our decision to make use of Internet indexing technology in a medical context was straightforward because of its ubiquitous presence, the available software components, and the maturity this technology has gained over the years. Three generations of automated Internet indexing systems can be identified as to their methods of compiling their data sets, their search interfaces, and their associated etymological metaphors and mythologies.[11] Firstly, the Archie search engine dating back to 1990 allowed for single keyword and regular expression searches of the file lists of anonymous File Transfer Protocol sites. Gopher, a distributed document network protocol preceding the World Wide Web,[12] featured a second generation; the Veronica system allowed for Boolean queries of directory information and filenames. Thirdly, search engines that compile searchable databases of information accessible via the World Wide Web are in wide use today, such as the popular Google.[13] Indexing technology slowly found its way into computer operating systems like Microsoft Corporation's Windows Vista (released in 2007) and Apple Corps' Mac OS X 10.4 (released in 2005).

Two aspects of Internet indexing technology could not be applied to our medical documents because of absent hyperlinking. The first was "spidering": robots crawling the web to search for new and updated material by investigating hyperlinks. However, this did not pose a problem because the RIS kept track of new and updated reports. Secondly, page-ranking algorithms could not use rules based on the number of referrals by other documents.

A nationwide study conducted by Vorbeck et al[14] in Austria, apart from the Alps, a country similar to Belgium, showed that radiologists were already familiar with the Internet in 1999. Today, all radiologists at our institution have a broadband Internet connection at home, including secure access to the electronic medical records and the PACS.

Scalability

Generally, the duration of building an index should be less than the desired timeframe in which new or updated documents show up in the system. Build times can increase beyond this point when document access speeds decrease or when the total size of the document collection grows. To counter this limitation, indexes can be built incrementally; however, care must be taken because some indexers do not recalculate frequencies of very common words for automatic exclusion. Similarly, indexes can be built in parallel; both mechanisms make use of index merging. Judging by our current results and the obsolete hardware used, we estimate that our indexing system scales up by two orders of

Table 3. Refining of a Query by a Typical User

Time	Query	Number of Results
12:00	cirrhose	528
12:02	cirrhose ascites	183
12:02	cirrhose ascites MR not angio*	100

magnitude, the main bottleneck being the non-distributed index.

To scale a cross-patient search engine, hospital-wide privacy considerations have to be taken into account. The hospital's policy on patient privacy should be implemented in strict access rules based on document metadata.

Semantics

A disadvantage of keyword-based searching is the simple fact that radiologists or researchers want to find the meaningful concepts behind their keywords. To make word indexing more or less suitable for concept searching, four search engine features are necessary. Firstly, because medical terms mostly consist of more than one word, it has to be possible to search for phrases. Page-ranking algorithms can help by giving a higher score to documents with adjacent keywords, but in practice, the sole use of this technique alone is not specific enough. Secondly, very frequent words with low significance can hamper phrase searching and should not be indexed; when "of" and "the" are excluded, the phrases "MRI of the brain" and "MRI brain" yield the same results. A third helpful feature is Boolean logic to facilitate binary operations like "and," "or," and "not" in combination with parentheses. Finally, a word-stemming technique can be used to find all words with a common root. Although we did not implement word stemming because our multilanguage set-up required building a separate stemmed index for each language, the use of wildcards turned out to be an adequate alternative. Word stemming is part of the broader method of fuzzy indexing, which can be used to find words with similar pronunciations.

To find conceptual information, users build their queries in multiple steps; hence, the user interface should allow for fast reviewing of results. Table 3 shows a typical user refining his query. Note that structured reporting, which uses standardized information concepts, does not have the semantic problems of free-text reporting.

A lot of research regarding medical semantics has been conducted, ranging from natural language processing techniques[15–17] to building unified medical lexica[18,19] for multiple languages.[20] The authors hope to combine the indexing solution with some of these scientific methods and technology regarding the semantic web[21] in the near future.

CONCLUSION

In this digital age, special care should be taken that the ever-growing amount of on-line information does not deteriorate into an inaccessible swamp of mere bits and bytes. On should strive to provide fast, clear, and easy access to information sources and, hence, increase the total value of stored data.

Designing and implementing an indexing solution for a large set of radiological reports and images turned out to be a technically challenging but educational endeavor. For research and training purposes, it certainly is a valuable and convenient addition to our radiology informatics toolbox. The use of open-source technology is highly recommended to reduce both implementation time and cost.

ACKNOWLEDGEMENTS

The authors would like to acknowledge Kevin Hughes for creating the original SWISH web indexing program and all contributors to this software's successor, Swish-e. Acknowledgement also goes to all authors of the other open-source software we used: Larry Wall for creating the Perl programming language,[22] Jef Poskanzer for his "thttpd" web server,[23] and finally the many researchers & coders of the FreeBSD operating system.

REFERENCES

1. Garfield E: Citation indexes for science. A new dimension in documentation through association of ideas. Science 122:108–111, 1955

2. Ziv J, Lempel A: Universal algorithm for sequential data compression. IEEE Trans Inf Theory 23(3):337–343, 1977

3. The FreeBSD project: FreeBSD. Available at http://freebsd.org. Accessed 27 May 2007.

4. McKusick M, Neville-Neil G: The Design and Implementation of the FreeBSD Operating System. Reading: Addison-Wesley, 2005

5. Health Level 7, Inc.: HL7. Available at http://hl7.org. Accessed 27 May 2007.

6. Google, Inc. Google Search Appliance. Available at http://google.com/enterprise/gsa. Accessed 27 May 2007.

7. Swish-e: Simple Web Indexing System for Humans—Enhanced. Available at http://swish-e.org. Accessed 23 May 2007.

8. Voet T, Duyck P: How to organise reporting: advice from the trenches. Imaging Manag 7:38–39, 2007

9. Voet T, Devolder P, Vercruysse J, Smeets P, Duyck P: Development and implementation of an indexing solution for a large set of radiological reports and images. Proc Radiol Soc N Am 90:457–458, 2004

10. SourceForge.net: Radiological Indexing. Available at http://sourceforge.net/projects/rad-indexing/. Accessed 9 July 2007.

11. Mcmurdo G: How the Internet was indexed. J Inf Sci 21 (6):479–489, 1995

12. Frana PL: Before the web there was Gopher. IEEE Ann Hist Comput 26(1):20–41, 2004

13. Brin S, Page L: The anatomy of a large-scale hypertextual web search engine. Comput Netw ISDN Syst 30(1–7):107–117, 1998

14. Vorbeck F, Zimmermann C, Vorbeck-Meister I, Kainberger F, Imhof H: Internet use in radiology: results of a nationwide survey. Eur J Radiol 31(2):141–151, 1999

15. Haug PJ, Ranum DL, Frederick PR: Computerized extraction of coded findings from free-text radiologic reports: work in progress. Radiology 174(2):543–548, 1990

16. Hripcsak G, Austin JHM, Alderson PO, Friedman C: Use of natural language processing to translate clinical information from a database of 889,921 chest radiographic reports. Radiology 224(1):157–163, 2002

17. Cooper GF, Miller RA: An experiment comparing lexical and statistical methods for extracting MeSH terms from clinical free text. J Am Med Inform Assoc 5(1):62–75, 1998

18. Humphreys BL, Lindberg DAB: The UMLS® project: making the conceptual connection between users and the information they need. Bull Med Libr Assoc 81(2):170–177, 1993

19. Johnson SB: A semantic lexicon for medical language processing. J Am Med Inform Assoc 6(3):205–218, 1999

20. Zweigenbaum P, Baud R, Burgun A, Namer F, Jarrousse E, Grabar N, Ruch P, Le Duff F, Forget JF, Douyere M, Darmoni S: UMLF: a unified medical lexicon for French. Int J Med Inform 74 (2–4):119–124, 2005

21. Berners-Lee T, Hendler J: Publishing on the semantic web. The coming Internet revolution will profoundly affect scientific information. Nature 410(6832):1023–1024, 2001

22. Perl.org: The Perl Directory. Available at http://perl.org. Accessed 28 May 2007.

23. ACME laboratories: Thttpd: tiny/turbo/throttling HTTP Server. Available at http://acme.com/software/thttpd. Accessed 28 May 2007.