Corpus Access for Beginners: the W3Corpora Project

Doug Arnold
Department of Language and Linguistics,
University of Essex,
Wivenhoe Park,
Colchester, Essex,
CO4 3SQ, U.K.

email: doug@essex.ac.uk

November 21, 2000

1 Introduction

This paper describes the results of an attempt to make Linguistic Corpus resources available over the World Wide Web, and provide novice users with both a theoretical and practical introduction to Corpus Linguistics: the W3Corpora Project.[1]

A linguistic "corpus" is a collection of texts, sometimes annotated with information at various levels of detail (e.g. about the gender of the speaker/writer, or about grammatical structures). Such collections are important in many areas of linguistics and related disciplines, and are becoming more important as more and larger corpora become available, and better techniques are

[1]This is a revised version of one chapter of the final project report (Arnold et al., 1999, Chapter 2). For a variety of reasons, that presentation is not ideal (the intended audience is rather narrow and specialized, the format is rather bulky, and there is a good deal of irrelevant material), and there has been a demand from a number of quarters for a more manageable general description of the project results and methods which gives more extensive discussion of the technical details than is possible in a conference paper or short journal article. I hope this paper may satisfy that demand while still being of some general interest.

The project was the joint work of several people, Ylva Berglund, Natalia Brines–Moya, Martin Rondell and the author (but I alone am to blame for this presentation, in particular, all the faults of style, fact, or interpretation are mine). The project was funded by JISC (the Joint Information Systems Committee of the UK Higher Education Funding Councils), under JTAP (the JISC Technology Application Programme), as part of project JTAP–2/247, which also involved the development of the "Internet Grammar of English" by a team at University College London, cf. (cf, Aarts et al. (1999)). and Arnold (1999) available from http://clwww.essex.ac.uk/~doug/.

Various other particular aspects of the project have been discussed in: Arnold (1997), Brines-Moya and Hartill (1998) , Arnold and Berglund (1998), and Arnold (2000).

introduced for manipulating them. The starting point for this project was the observation that despite this importance, the use of corpora was less widespread than it should be, and that the main reason for this was the difficulties that faced the newcomer in getting started. This is true whether the newcomer is a student or an established researcher. The difficulties are of several kinds: linguistic corpora tend to be very large (too large to fit on the size of disk that one typically found on a normal PC two years ago), and the software required to manipulate them can be difficult to obtain and install, especially for the computational naive user. The user must then familiarize himself or herself with the software, and then decide what to do with the software. (Bear in mind that this is a newcomer — someone who has perhaps heard that linguistic corpora are important and believes they must be useful in their studies or research, but does not know exactly how to exploit them).

The aim of the project was to provide free access to existing linguistic corpora via the Word Wide Web (WWW) to students and researchers in Linguistics and related disciplines, together with programs that would allow them to use the corpora, and with 'help' and tutorial pages that would show them how to use them for various tasks. The idea was that a new user would need only a Web connection and a browser; beyond this, no investment of money, and little investment of effort would be needed: there would be no need to obtain and install corpora, or download and install software, and the interface to the corpus manipulating tools would already be familiar (since it would be based closely on their web browser).

The intended audience for this resource was thus mainly newcomers or 'casual' users.[2] Of course, one would hope that experienced users of corpus resources would find something useful, but realistically one would expect experienced users to have corpora and tools locally available, and one would expect a newcomer who finds corpus resources useful to invest in acquiring and installing corpus resources and tools on their local system. One implication of this is that accessibility and ease of use is more important than sophistication or power. A further implication is that one should provide not just the tools, but a considerable amount of discussion of how the tools can be used, and a general discussion of the whole context of Corpus Linguistics.

Considerations such as this lead us to formulate the following desiderata for the system:

- The system should be immediately usable by anyone with WWW access and a Web Browser, for example:
  - it should be usable without the need to install or download any programs;
  - it should be usable without the need to register and get authorization.
- The interface should be as 'friendly' and easy to use as possible; it should be supported by extensive on-line help, and backed up by detailed information about Corpus Linguistics in

[2] An example of an experienced casual user would be someone who teaches a course which touches on Corpus Linguistics but who does not want to go to the time or trouble of setting up student access and network installation of other corpus manipulating software, which might seem pointless if all one wants is to expose students for a week or two.

general, and how to 'do' Corpus Linguistics in a practical way, using a tool such as this system.

- It is typical of novice users that they make mistakes with queries; thus, there should be some method for users to correct and 'refine' their queries very easily (this lead us to the idea of an editable 'search history', see below).

- It should be possible for a user to search their own Corpora — in this way a user can explore not only what is possible in general, but what is possible in relation to the kinds of material they are interested in or have to deal with.

To these we added the further requirement that the source code should be freely available (in GNU 'Copyleft' style).[3] This is partly to allow the system to be installed on other sites (for example, an experienced user might want to eliminate the network overhead by installing the system locally).

We had originally hoped it might be possible to use existing software to perform corpus searches, but this turned out not to be the case. This meant that in addition to designing and implementing a WWW interface, a 'search engine' engine also had to be implemented, giving essentially three tasks: (i) search engine; (ii) WWW interface; (iii) on-line help and documentation. In the event, for reasons we will go into below, we quickly came to the conclusion that (iii) should contain far more than just information about this particular system, and that it should contain information about Corpus Linguistics in general, and a practically oriented tutorial on using this system for some typical tasks where Corpus Linguistics techniques are useful.

It is thus useful to think of the project as having essentially three parts:

1. A Search Engine and WWW interface, that allows users to formulate Corpus Queries, and see search results;

2. A tutorial about doing Corpus Linguistics in general, and using this system in particular.

3. A collection of pages giving Information of general interest about Corpus Linguistics.

Taken together, these provide an extensive introduction to Corpus Linguistics for the newcomer, who can gain insight at both practical and theoretical levels.

The remainder of this paper is structured as follows. Section 2 gives an overview of the system from the users point of view, and describes the basic functionality. Section 3 gives a brief overview of systems with similar functionality. Section 4 describes the programs that provide the interface and perform searches. Sections 5 and 6 describe respectively the pages that give information of general interest about Corpus Linguistics, and the tutorial. The corpus resources that are available are described in Section 7. Section 8 provides a conclusion, including some evaluation and general reflection.

[3] See, for example `http://www.gnu.ai.mit.edu/`.

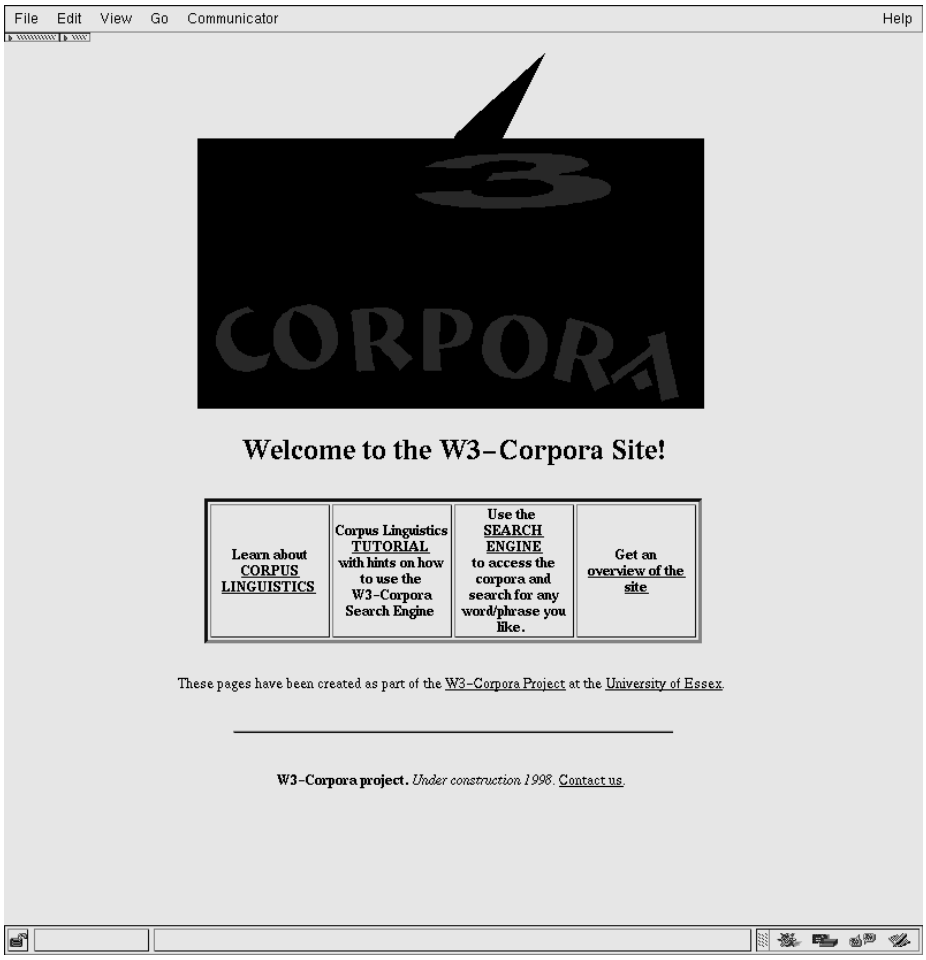

Figure 1: Top Level W3Corpora Page: main entry point to the site.

2 Overview: from the users point of view

On entering the W3Corpora Pages at the top level, the user is presented with a screen something like Figure 1, which offers four choices, allowing the user to:

1. Learn about Corpus Linguistics (see Section 5: Information Pages);
2. Follow a Tutorial about Corpus Linguistics, including hints on how to use the search engine (see Section 6: Tutorial);
3. Use the search engine itself (see Section 4: the Search Engine);
4. Learn something more about the W3Corpora project (to access some documents about the project, notably a description of the structure of the site, and a copy of this report — this part of the site is not discussed further in this report).

Before we discuss the first three of these, we will describe how a user can carry out a search, thereby describing the interface to the search engine.

If from the initial W3Corpora page the user opts to use the search engine, they are asked whether they want to access all available corpora (**full access**, which may require a registration process), or only the publicly available corpora (**limited access**). This allows the site to support both freely available and restricted access corpora. [4] Apart from checking authorization and determining which corpora are to be available to the user, the behaviour of the system is identical, so from now on we will ignore this, and assume the general case where the user accesses only publicly available corpora. On selecting **limited access**, the user is presented a screen like Figure 2, which allows them to specify a query by:

1. Selecting a Corpus — that is, a collection of texts to search;
2. Selecting a Search String — that is, specify what they want to search for;
3. Submitting the query.

In the first two cases the user is presented with a complete new screen containing a form to fill in, making their selection.

At the bottom of the screen is a button that allows the user to submit comments or suggestions. At the top of this (and every screen relating to corpus searches) there is a ‘help’ button, which generates a help message appropriate to the user at this stage of formulating a query or examining results. Typically, this describes what sorts of action the user is expected to carry out, and what any specialized terms mean.

[4] The fact that we have to restrict access to some corpora to only registered users reflects a condition imposed by the administrators of particular corpus resources.

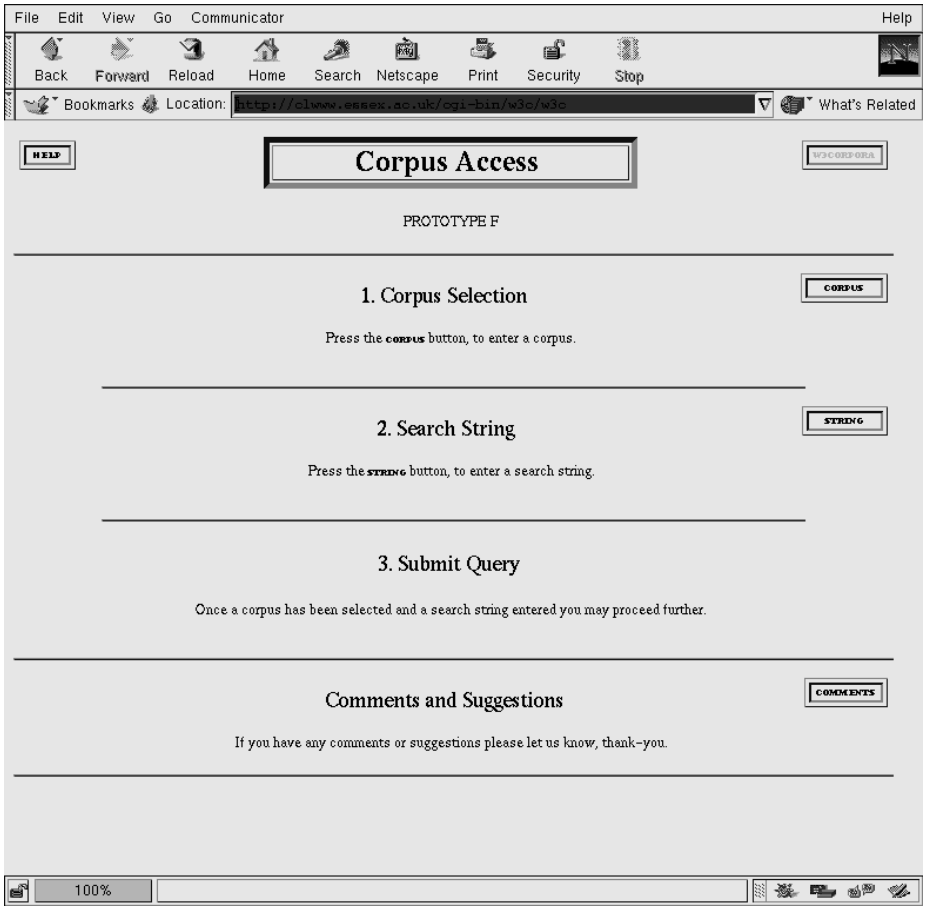

Figure 2: Top Level Search Page: Specifying main Search Options.

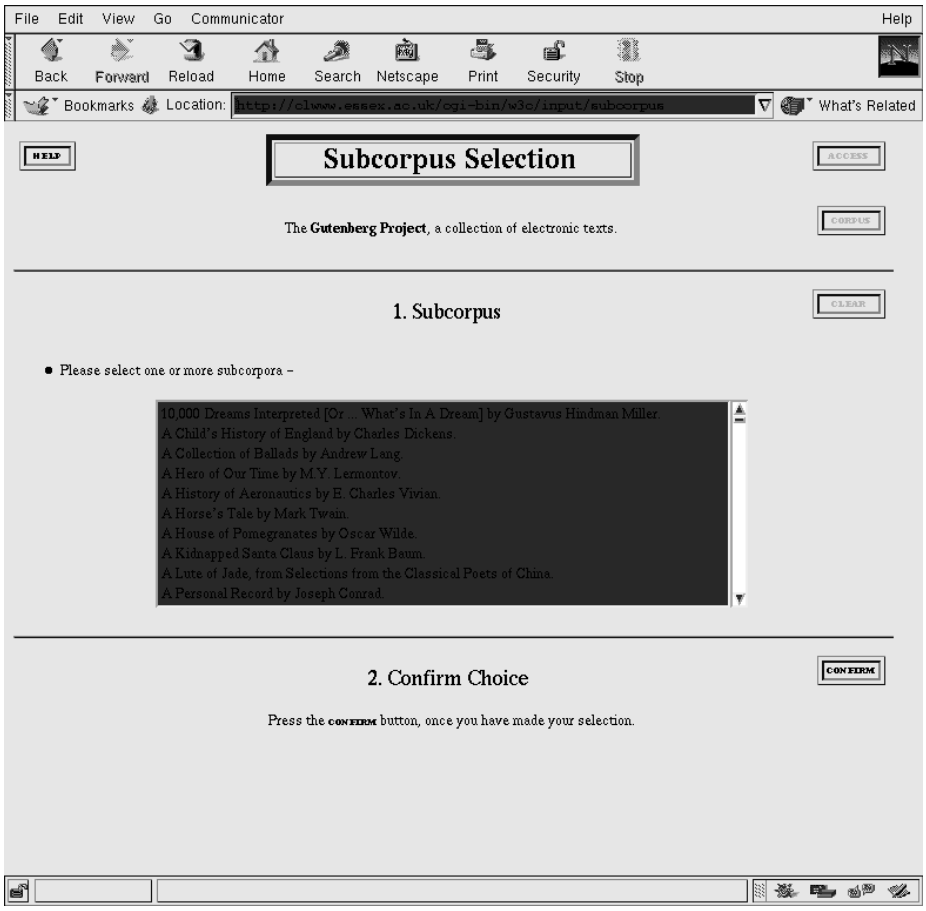

Figure 3: Selecting a Corpus.

In selecting a corpus, users are presented with a list of the available texts, from which they can select as many as desired (see Figure 3). When their choice is confirmed, they are returned to the main Corpus Access page, to specify a search string (see Figure 4). This involves specifying a *type* of search, and a *pattern*. The pattern is typically a regular expression. Providing different types of search essentially pre-defines certain common kinds of regular expression: types of search include searches that only succeed at the beginning or end of words, or which are anchored at both beginning and end (i.e. whole words), as well as general regular expressions. When this choice is confirmed, the user is again returned to the top level page, from where they can submit the search/query.

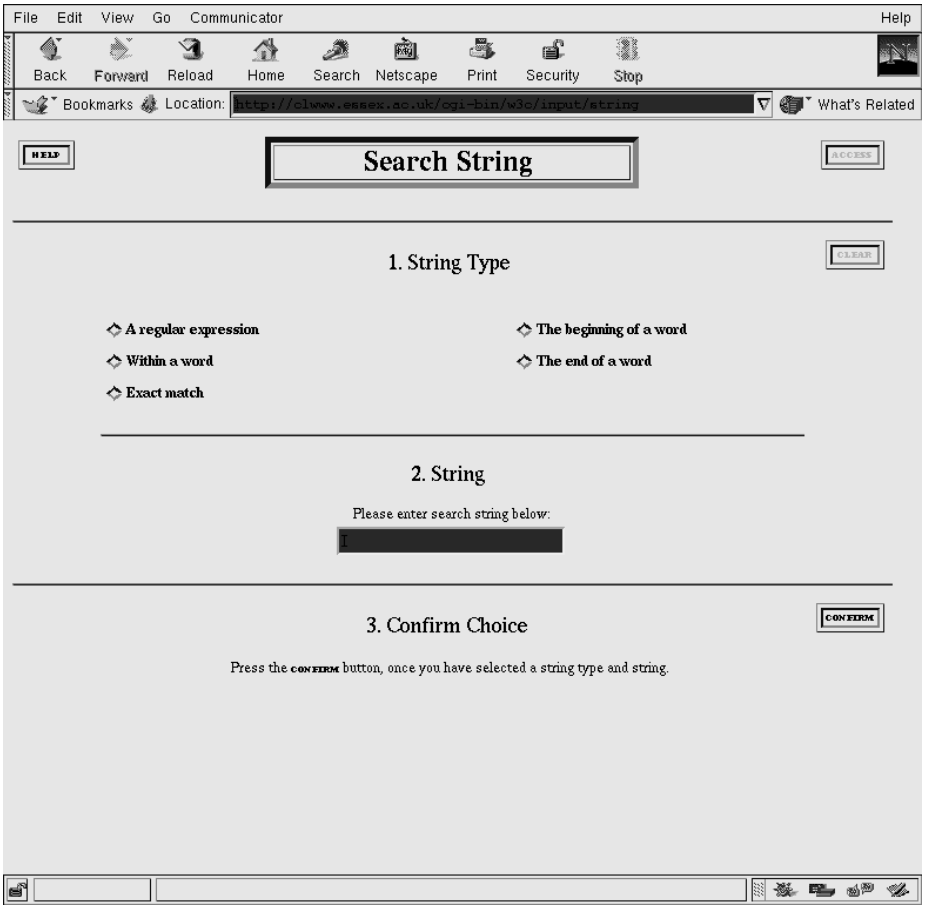

Figure 4: Specifying a Search String.

Suppose, for concreteness, the user chooses the Gutenberg corpus, selects the following documents to form the sub-corpus, and opts for a search for the regular expression `[Nn]ice` — that is, the string “nice”, possibly beginning with an uppercase:

> [Several] Works by Emile Zola - Nana, Miller’s Daughter, Captain Burle.
> War of the Classes by Jack London.
> A History of Aeronautics by E. Charles Vivian.

Tomorrow by Joseph Conrad.
Tom Swift and his War Tank by Victor Appleton.
Travels with a Donkey in the Cevennes by Robert Louis Stevenson.
Under the Andes by Rex Stout.

Figure 5: First Search Results, for regular expression search [Nn]ice ("nice" possibly starting with an upper case letter).

Initial results of the query are returned as in Figure 5. By default, the first search results shown relate to frequency, and give the simplest information about number of matches. It is also possible to view how the results are distributed across the different parts of the sub-corpora (Figure 6), or to see the numbers of different words that were matched (Figure 7 — notice that matches were returned for *Venice* and *cornice*, which may or may not have been intended).[5]

Neither of these screens is very informative about how the word "nice" is used, however. For this, the user should click on the "DISPLAY" button at the top of the screen, which will generate a KWIC (Key Word In Context) display, as in Figure 8.

[5]Perhaps the user really wanted words related to *nice* — the search term should have been ^[Nn]ice, which only matches at the start of words. The user could also have required matches to be anchored at the start and/or end of

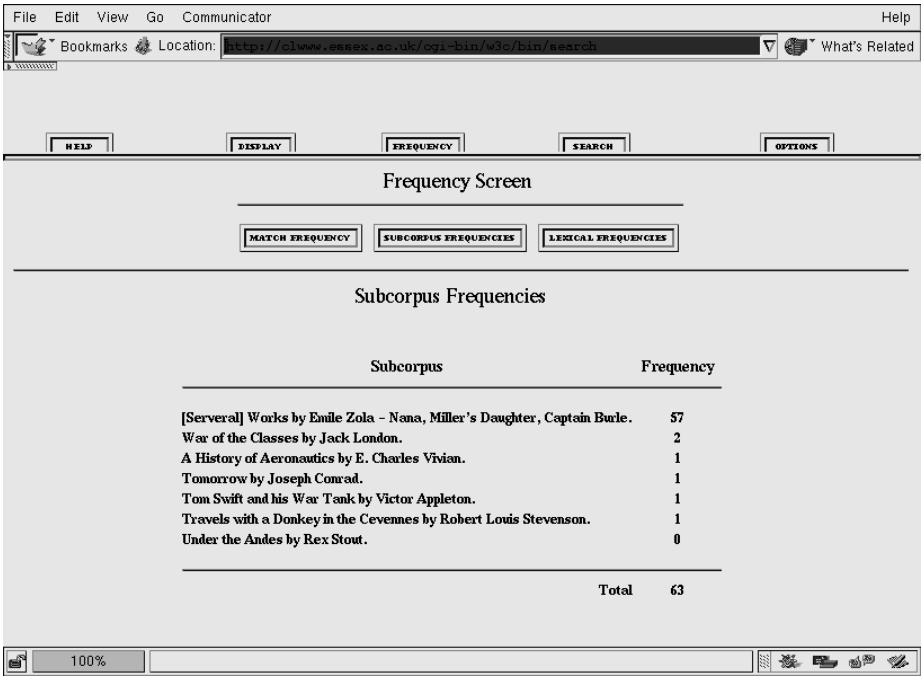

Figure 6: First Search Results, indicating sub-corpus frequency (for regular expression search `[Nn]ice`).

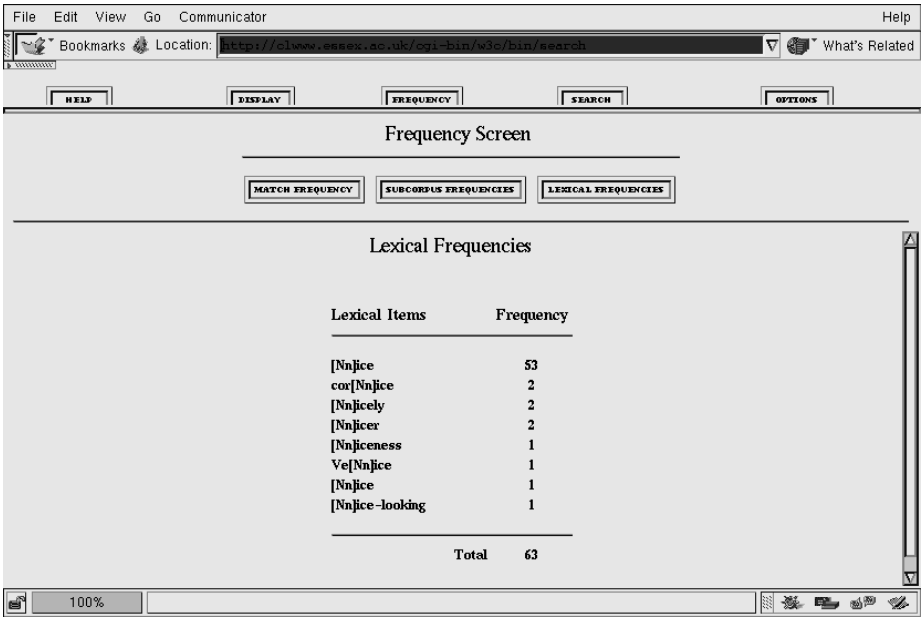

Figure 7: First Search Results, indicating Lexical Frequencies (for regular expression search `[Nn]ice`).

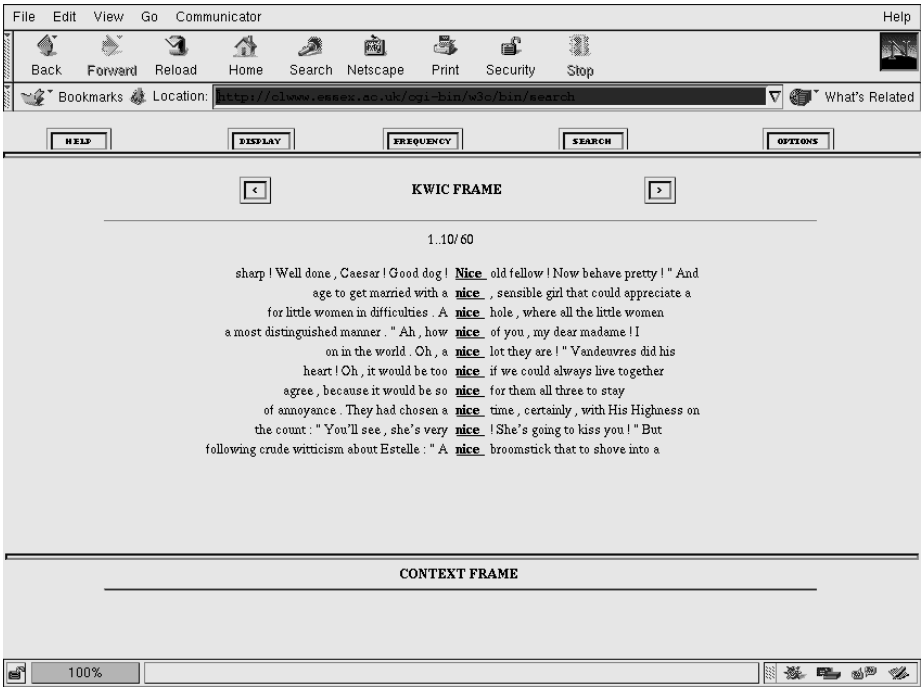

Figure 8: KWIC Display of first 10 search Results (for regular expression search`[Nn]ice`).

Clicking on the keyword gives the wide context, as in Figure 9, which also shows the effect of setting two other KWIC options, requesting the ability to DELETE items from the KWIC results, and requesting display of the key-reference of examples to the left of the KWIC entry. Clicking on the latter gives the source of the example below the KWIC index itself.

At the top of the screen several other buttons are to be seen. These behave as follows, and give an idea of the range of possible actions and options available.

HELP gives an appropriate help page (e.g. from the top level search screen, this is just a description of what the various buttons do, and what various terms mean).

DISPLAY generates a display which shows the actual hits matching a search: The DISPLAY page is divided into three sections:

KWIC: Concordance with one hit per line. The user can configure a number of parameters, including: number of words on either side of match, how many matches to display, and in what order, whether to include punctuation, whether to give the source, and whether to allow the possibility of deleting particular search results from from the display.

KEY: Give information about the source of the hit.

CONTEXT: Give the larger context for one hit at the time.

---

the pattern.

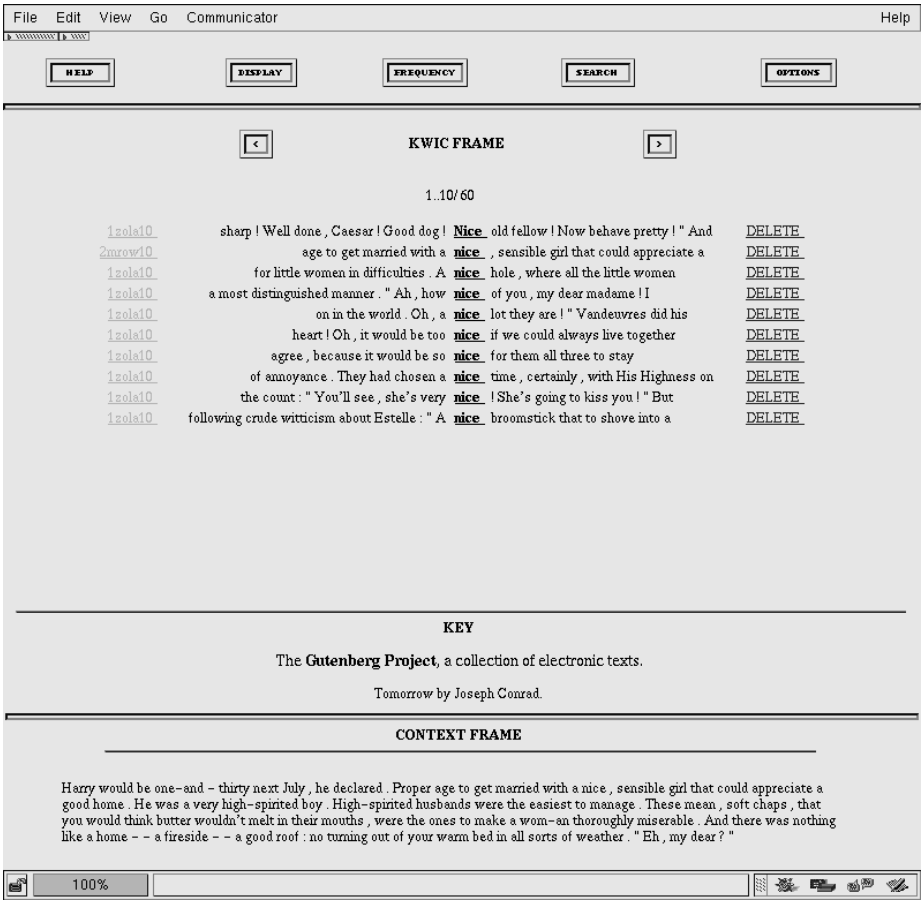

Figure 9: KWIC Display of search Results: the user has selected options which allow results to be deleted and which indicate which sub-corpus each hit comes from. At the bottom of the page the wider context of one of the hits is displayed (the user has clicked on one of the individual hits to obtain this).

FREQUENCY generates a page which shows how many hits there are matching a search: there are three further options here:

Frequency provides general information about hits (string, string type, corpus, total frequency).

Sub-corpus Frequency shows the number of hits per sub-corpus.

Lexical Frequency displays the frequencies of the various lexical items found by the search.

SEARCH permits the user to make a new search, refine the search, or see the *search history* (see below). There are three options available from the SEARCH page:

New Search (make new search with possibility to retain previous options).

Refine Search (return a subset of the hits from a previous search).

View Search history (display, and possibly edit, a list of previous related searches).

OPTIONS generates a form which permits the user to change the default options for how the hits are displayed:

General Options Choose whether to display hits from certain sub-corpora only. Choose whether the initial screen should contain Frequency or KWIC results.

Display Options Set options for how the hits are shown on the Display pages (context, markup, etc).

Frequency Options Set options for the Frequency pages (display order, percentage or raw figures).

Search Options Set options for New Searches.

This description is intended to give the reader an overview of the resources provided, and an idea of the basic functionality of the search engine and interface. It does not describe all the features, and we will not try to do that here. However, two other features are worth mentioning: the possibility of searching users' own corpora, and the search 'history' mechanism.

As well as searching the corporal that are provided at the site, it is possible for users to upload (by FTP) their own texts and search them (step by step instructions on the use of FTP for this purpose are given when help is requested on the page dealing with user-defined corpora). The choice of a user-defined corpus is available when the user selects the corpus to search. What happens is that when a user selects 'User Defined' corpus (in place of, e.g. Gutenberg), files that have been uploaded in this way are processed for searching, and then searched in the usual way — the full functionality of the search engine is available (the only difference is the information about the corpus that is displayed.)[6] The idea is that a user can explore whether the searching techniques are useful on their own material.

[6]Only plain text corpora can be handled as 'User Defined' Corpora in this way.

The ‘history’ mechanism allows a user to create and edit a history of searches. Suppose that after searching for “[Nn]ice” a user decides that what they really wanted was “^[Nn]ice” (which matches only at the start of a word). One possibility is to edit the search results directly, deleting the unintended hits. However, an alternative is that they choose to ‘refine’ their search, that is, to define a search over the previous hits (the strings matched by the previous search). When they have specified the new search string (e.g. “^[Nn]”), they will be shown a screen like Figure 10. This shows the history of the original search and its refinement(s), with information about the search string, frequency, etc. With this, they can choose which set of results they want to display. They can also edit the history by removing one set of results (cf. the Delete button).

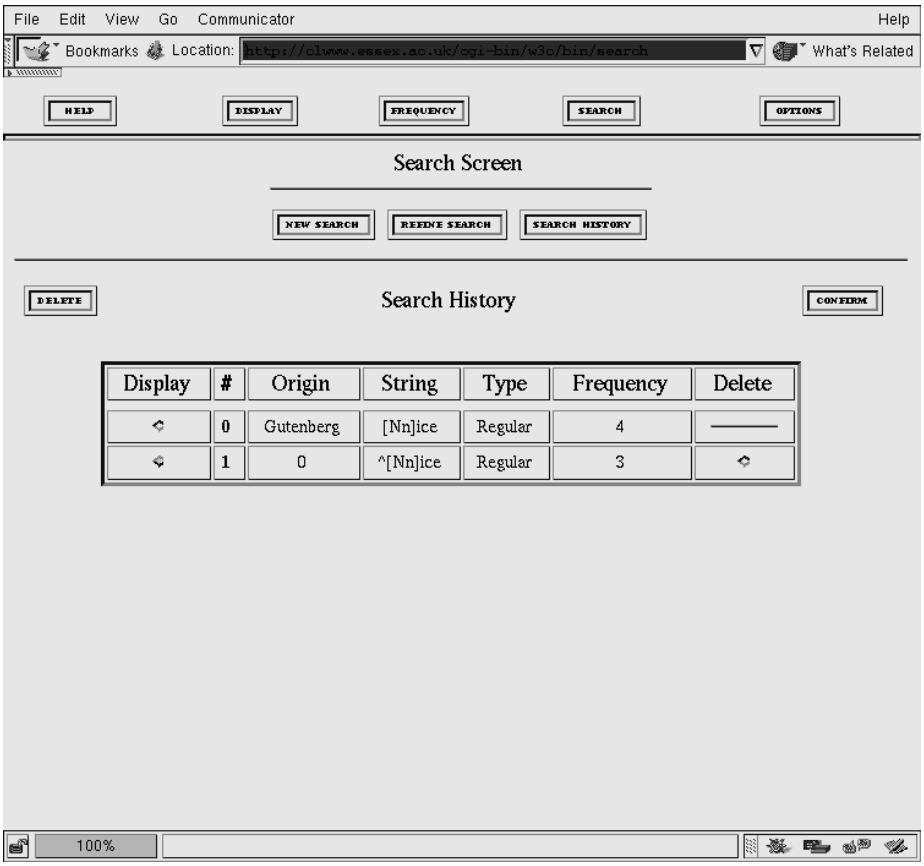

Figure 10: Search History screen: the user can try variations on a particular search, creating a ‘history’; they can move backward and forward through this history, displaying results. It is also possible to delete items from the history.

3 Alternatives and Existing Approaches

There are a large number of tools and systems that offer something similar to what the W3Corpora site seeks to provide. There are too many for us to discuss all of them in any detail (indeed, there

are too many for us to claim that we know about all of them), but it seems that none is exactly comparable. In the main, they cannot provide the kind of general, free, 'introductory' service that we have tried to provide, because they are commercial, and/or because they require downloading of software, and/or because they are platform specific. So far as we know, in no case is the source code freely available. Where they do provide semi-introductory access (e.g. by means of free registration and/or a guest account), there is generally very little in the way of of tutorial material.

Among the tools and systems available for corpus analysis in general, the following are worth noting:[7]

ptx Traditionally, Unix-style systems provide a permuted index generation program, i.e. a facility for performing corpus searches producing KWIC output. 'Ptx' is the GNU version of this. It is available for Unix, Dos, and Macintosh (but requires a C-compiler). It is quite flexible, but has essentially no user interface (interaction is by standard Unix-style command line and command line options, output goes to Standard Ouput). The GNU version can handle multiple input files at once, but does not handle input files that do not fit in memory all at once (this means, for example, that searching all the Gutenberg texts available at the W3Corpora site is well beyond what is possible on a system with normal memory and with normal load). It is widely available from standard archive sites, and easy to install (at least on a modern Unix system).

Conc 1.7 Concordancing program, available from the Summer Institute of Linguistics. Software must be downloaded and installed locally. It is only available for Macintoshes, and suitable only for relatively small files.[8]

Corpusbench Supports a wide variety of searches (frequency counts, concordancing, simple grammatical and morphological analyses), and collocational information (via Mutual Information scores). This is a commercial product available from Textware Direct, Horscholmsgrade, 20 2 DK-2200, Copenhagen, Denmark.

ICEUP The International Corpus of English Utility Program. This is for use only with the International Corpus of English, it is intended for Windows and it is not currently web accessible.

LEXA Tagging and concordancing software, distributed by ICAME (Norwegian Computing Centre for the Humanities). Must be installed locally.[9]

ParaConc ParaConc is a bilingual/multi-lingual concordance program (in different formats) designed to be used for contrastive corpus-based language research. Macintosh only (but a

[7] It is worth stressing that the information given here was accurate at the time the systems were reviewed, but it may easily be out-of-date. Most of the system described are under development, which normally means that they either improve, or become free or are ported to new platorforms. The web accessible sites listed here are all referenced via the project web-pages. Useful information about Macintosh and DOS based corpus analyais tools can be found at `http://info.ox.ac.uk/ctitext/enquiry/`.

Windows version is under development). Must be installed locally. It is restricted to individual users (site use requires purchase of a license).[10]

LDB Nijmegen Linguistic Database Software: for use specifically with the Nijmegen Corpus; must be installed locally.

Wordsmith Tools Supports word lists, concordancing, and text alignment. DOS/Windows only. Wordsmith Tools Version 2 is a commercial product, but version 1 can be downloaded free of charge.[11]

MicroConcord Word counts, concordancing, simple syntactic and morphological analysis; works with output from a variety of word processors; a commercial product.[12]

IMS Corpus Workbench Supports KWIC concordances, frequency counts, multi-lingual concordances from aligned corpora, and provides a query history. There is an X- (specifically Motif-) based graphical interface (xkwic). Avalailable for Unix (Solaris and Linux); requires license registration and local installation.[13]

Micro-OCP Concordancing, word lists, frequency lists and some statistics: DOS/Windows only; requires local installation.[14]

Multiconcord A Multi-lingual Parallel Concordancer for Windows, being developed at the University of Birmingham under a Lingua project to develop a Windows-based parallel concordancer for classroom use. It is not intended for use over the WWW.[15]

Multext The Multex project is developing a series of tools for accessing and manipulating corpora, including corpora encoded in SGML, and for accomplishing a series of corpus annotation tasks, including token and sentence boundary recognition, morphosyntactic tagging, parallel text alignment, and prosody markup. Annotation results may also be generated in SGML format. Tools are under development, upon completion, all tools will be publicly available for non-commercial, academic use.

Sara Sara is the server for searching the British National Corpus. It assumes special Windows/Dos software is installed on client machines. Registration and licensing are required (see below, where we will say something about the Web-accessible version of this).

Several of these systems are only under-development, for the rest, one sees that many of the products are only available commercially, and/or require some registration or licensing process

[8] `http://www.sil.org/.`
[9] `http://www.hit.uib.no/icame/icame.html`
[10] `http://www.ruf.rice.edu/~barlow/parac.html`
[11] `http://www1.oup.co.uk/elt/catalogu/multimed/`
[12] `http://www.nol.net`
[13] `http://www.ims.uni-stuttgart.de/projekte/CorpusWorkbench/`
[14] `http://www.oup.co.uk/`
[15] `http://sun1.bham.ac.uk/johnstf/lingua.htm`

which may be enough to deter the uncommitted user. One can also see a wide range of functionality. However, as regards their post-installation functionality, four systems: — *ptx*, *Sara*, *Micro-OCP*, and the *IMS Corpus Workbench* — are particularly relevant, since they provided the starting point for our work (see below).

At the time the project started, there were no sites providing WWW access to corpora. However, over the life of the project, a number of sites have emerged offering some kind of web access.

BNC The British National Corpus is a very large (over 100 million words) corpus of modern English, both spoken and written. The BNC site provides access to a subset of the British National Corpus on a trial basis.[16]This permits simple searches on-line, but with limited number of hits, limited information about the hits. Registration for a trial account (20 days) is required.[17]Full access requires downloading (Windows) client program (available for Windows95, and Windows3.x only), and payment of an annual registration fee. It is restricted to users within the EC.

Cobuild This site gives limited access to the Cobuild Corpora: the "Bank of English"(over 50million words).[18]The page is intended to provide a flavour of the kinds of search that can be carried out. It is possible to search for regular expressions (including a special character which matches inflectional endings), combinations of words, and part of speech tags. Only 40 lines of concordance are returned, and no information about frequency, or wider context is accessible. It is also possible to search for collocates of words, based on either of two statistical scores (mutual information and T-score), ranked by statistical significance (100 collocates are returned by default).

The site does not provide much in the way of help pages, and there is no tutorial.

Bergen Corpus of London Teenager Language At this site it is possible to search a pilot version of the Bergen Corpus of London Teenager Language corpus using the TACTWeb software.[19]TACTWeb is intended to make TACT software usable over the WWW, i.e. to make a TACT style text database accessible over the WWW. [20]This is very close in intention to the W3Corpora project. At the time of writing, it is still under development.

Canadian Hansard This site permits access to the proceedings of the Canadian Parliament in English and French.[21]These are parallel corpora (English and French), searches may be mono- or bi-lingual (in either case, the results returned are bi-lingual — i.e. the user sees both the context where the search term appears, and translation):

- with the simple query interface, entering a word or expression in one language will retrieve examples of its use together with the translation of these examples. For instance, typing *passer un sapin* will allow the user to to see how this expression is used and how it can be translated.
- With the bilingual query interface, the user can also enter a pair of words or expressions to retrieve examples where one element is translated as the other. For example, entering *commitment* in the English Expression field and *attachement* in the French

Expression field will produce examples where one of these words is translated as the other.

Normally, the program searches for expressions verbatim: a query like *pull the plug* will find all occurrences of that string (and none of *pulled the plug* or *pull the plugs).* It is also possible to perform a dictionary search, e.g. the query: *pull+ the plug* will look for *pull the plug*, but also *pulling the plug*, *pulls the plug*, etc, and searches for words that do not appear contiguously (e.g. *make ... arrangements*); there is also a restricted form of ellipsis (indicated as "..") which only spans a few words (25 characters). All searches are case-independent.It is possible to view the wider context of search results. No frequency information is provided.

Old English Corpus This site gives access to a corpus of "all surviving OE material" — since it is not intended for (Modern) English, this is not strictly comparable with the current project.[22]It supports simple, boolean, proximity, and bibliographic searches (simple searches, are for whole or parts of words; boolean are for boolean combinations of several words (and/or/not); proximity searches permit searching for words within a certain distance of each other — e.g. within 80 characters; bibliographic searches are for works by individual authors).

Swedish Government Site This site gives access to "Regeringsforklaringen":[23]the yearly declaration of the Swedish government issued in Swedish, English, French, German, and Spanish. Simple searches are supported. At the time of writing, this is simply a demonstration program.

LDC/Brown Corpus Text Corpora,[24]and Speech Corpora,[25]are available via the Linguistic Data Consortium.[26]It is possible to access the Brown Corpus (1 Million words of American English) remotely, after registration at the Linguistic Data Consortium site. [27]For individuals who are not (affiliated to) members of the LDC it is possible to register as a guest (and later upgrade to full membership), and access corpora with the password that is supplied; authorization and password are sent to the user by email. Frequency information is available, and a wide variety of searches is supported, corcordances can be generated, and collocational information retrieved. Access to the TIMIT Speech Corpus is similar.[28]

It is obvious that some of these sites provide functionality that is not available at the W3Corpora site — notably (i) multi-lingual searching and searching over parallel corpora, (ii) collocational information, and (iii) 'dictionary style searching'[29] — and several provide access to far more extensive corpus resources.

On the other hand none of these sites duplicates what is available at the W3Corpora site. In particular, none of them provides the balance of easy (immediate) access to usable quantities of corpus material, with easy, customizable functionality, and extensive user support and tutorial facilities.

Based on a study and extensive use of several of the systems (*Sara*, *Micro-OCP*, the *IMS Corpus Workbench*, and *ptx*) a list of desiderata was drawn up for the facilities a user interface to corpus searching software should provide (cf. Brines-Moya and Hartill (1998)). Here we will summarize the main points. It will be seen that, with very few exceptions, the W3Corpora interface satisfies them.

Concordances The user should be able to specify the number of items (e.g. words) of context to be displayed on either side of the key word ('KWIC display flexibility'). It should be possible to access 'extended' context. It should be possible to search output, reorganizing, reducing, or constraining the search. The W3Corpora site provides all these.

Sub-corpora There are a number of reasons why dividing corpora into sub-corpora is useful (e.g. it makes it possible to compare searches across sub-corpora; to focus search on sub-corpora that contain interesting phenomena). The W3Corpora site supports this.[30]

Refining It should be possible to refine the output of a search: the W3Corpora site supports only one kind of refinement, where the text matched by the original search term is taken as the space for a further search. (Desirable additions might include the ability to refine the search by stating constraints on the surrounding context, or to use Part of Speech information).

[16] `http://thetis.bl.uk/lookup.html`.

[17] Full registration is available at `http://info.ox.ac.uk/bnc/sara/index.html`

[18] `http://titania.cobuild.collins.co.uk/form.html`.

[19] `http://www.hf.uib.no/i/Engelsk/COLT/index.html`.

[20] TACT is a text-analysis and retrieval system for MS-DOS that permits inquiries on text databases in European languages. It developed out of a collaboration between IBM and the University of Toronto in the 1980s.

[21] `http://www-rali.iro.umontreal.ca/TransSearch/TS-simple-uen.cgi`

[22] `http://www.hti.umich.edu/english/oec/about.html`

[23] `http://strindberg.ling.uu.se/~corpora/rf/`

[24] `http://www.ldc.upenn.edu/ldc/catalog/html/text.htmlText`

[25] `http://www.ldc.upenn.edu/ldc/catalog/html/speech.html`

[26] `http://www.ldc.upenn.edu/ldc/about/index.html`

[27] `http://www.ldc.upenn.edu/ldc/register.html`

[28] `http://www.ldc.upenn.edu/readme_files/timit.readme.html`

[29] That is, the ability to search for inflectional variants of a word — to use *take* as the search term and recover instances of *takes*, *taken* and *took*, or for a regular verb like *walk*, to recover *walks* and *walked*. However, such searches can be easily simulated using regular expressions.

On the other hand, a 'search history' facility is provided which allows a user to explore variations on a basic search pattern, and compare results across the variations.

Saving/Editing The ability to save, and possibly edit results is useful. The W3Corpora site is generous in terms of saving results (for each search session, results are saved for over a month, though they cannot normally be accessed them once a user has terminated a session, either by starting a totally new search, or killing their browser). But it is not possible to edit the results of searches: this would require providing full editing facilities across the Web, which was beyond the scope of the project. However, the user can save their search results using normal drag and drop from the browser windows into a text editor of their own (and they can then re-submit them as a user-defined corpus, if they wish).

Query Syntax There should be a balance between user-friendliness and flexibility/power: standard options (e.g. search only for whole words) should be available directly, but the full power of regular expressions should also be available (W3Corpora permits the user to construct 'simple' searches directly, as well as giving access to the full power of regular expressions).

Availability and Installation

- It should be easy to install new corpora (in the case of the W3Corpora site, this is simply a matter of uploading the files, and starting a search);
- It should be possible to install the system on a variety of platforms, with appropriate (clear, detailed, etc.) instructions. In the case of W3Corpora, the user is not required to undertake any installation. It is accessible from any system that is connected to the WWW and supports a web browser.

Other Features

- It is important that results are displayed in a clear way — e.g. layout and colours should be used in such a way that it is easy to pick the key words out in a KWIC display.
- Help pages and tutorial pages should be readily available, well-structured, and appropriate.
- It should be possible to modify and adapt the way the output is presented.

To conclude: no existing system or site offers the functionality of the W3Corpora site. It is unique in terms of ease and flexibility of access, and the quality of the user help and tutorial material. Moreover, though there are sites that offer more in terms of functionality, and size and variety of corpus resources available, it compares well to most other sites and systems in these terms too.

[30]The only desirable feature that is not supported is the ability to form a sub-corpus from (say) all the paragraphs which contained hits in a previous search. We felt that permitting users the to collect parts of corpora in this way would be potentially disasterous from a copyright point of view. Of course, the W3Corpora site permits the user to collect their own subcorpus by other means, upload it and search it .

4 Implementation of the Interface and the Search Engine

This section is mainly concerned with the scripts that provide the user interface, and perform the actual searching, but a few words about the directory structure of the 'static' web pages, which provide the information and tutorial, will also be useful.

The web-server which provides these pages is `http://clwww.essex.ac.uk`. The software which supports this is to be found in directories as given in Table 1 (specified relative to the `http` 'root').

`/cgi-bin/w3c/`	*main scripts, described in the rest of this Section*;
`/cgi-bin/w3c-passwd/`	*scripts for dealing with access requiring authorization*;
`/htdocs/w3c-ige/`	*joint web page for the entire W3Corpora/IGE project* ;
`/htdocs/w3c/`	*'static' web pages — see Sections 5 and 6*;
`/htdocs/w3c/general_info/`	*general information about the W3Corpora part of the project*;
`/htdocs/w3c/corpora/`	*information about the corpora that can be searched*;
`/htdocs/w3c/corpus_ling/`	*the 'information' pages (Section 5)*;
`/htdocs/w3c/help/`	*text for on-line help messages*;
`/htdocs/w3c/registration.html`	*form for user registration* ;
`/htdocs/w3c/index.html`	*main access point/starting place for the whole site;*

Table 1: Top-level directory for the whole web-site (excluding cgi-scripts).

The interface and the search engine that carries out corpus searches are implemented as cgi-scripts, using Perl (version 5). As is customary, the scripts are in a subdirectory of `httpd/cgi-bin/`, namely `http/cgi-bin/w3c`. The directory structure at this level is given in Table 2.

The full source code for the system totals over 12,000 lines of code, much of it very straightforward and of little intrinsic interest (being concerned with generating html for display purposes), and is not discussed here (most of it can be found in (Arnold et al., 1999, Appendix B).

Many aspects of the system are controlled by definitions in `cgi-bin/w3c/bin/header` (see the code appendix) in particular, this script sets up variables whose values specify the directories where various things can be found.

4.1 The Search Engine

The techniques used for corpus searching are fairly standard. In this section we will try to give a flavour of how they work (full code listings are given in an Appendix to this Report). They are most easily understood by starting with how corpora are prepared for searching. We will describe this in relation to a toy corpus of about 20 words (Figure 11). The scripts involved in creating and searching corpora are listed in Table 3, and Table 4.

cgi-bin/w3c/	
w3c	*the top level script — see Section 4.1;*
display/	*contains scripts relating to the display — Section 4.2;*
corpus/	*contains the actual corpora — see Section 4.1;*
bin/	*contains the main searching programs — see Section 4.1;*
results/	*contains results of individual searches — see Section 4.1;*
input/	*scripts to generate forms by which the user specifies initial corpus and search string (Section 4.2);*
lib/	*library routines — see Section 4.1;*
comments/	*scripts for dealing with users' comments — Section 4.3;*
registration/	*scripts relating to registration for restricted corpora — Section 4.3;*
help/	*contains scripts to generate on-line help messages — Section 4.3;*

Table 2: Source code top-level directory structure.

cgi-bin/w3c/bin/	
header*	*sets up all global variables, and default values for user variables*;
search*	*called by the top level script* `w3c`*, sets up displays, and calls* `locate`;
selection*	*displays list of search-able corpora for user to select*;
panel_select*	*generates html frame to contain the list of search-able corpora*;
search_help*	*displays an error message if something serious goes wrong (should not normally be seen by users)*
search_help_title*	*generates basic outline for help screen*;
locate*	*the main scripts finding matches for search terms in indexed corpora*;

Table 3: `/cgi-bin/w3c/bin:` Searching and corpus preparation scripts

cgi-bin/w3c/lib/	
corpus*	*sets up various general arrays (e.g. associating file names with descriptions of the corpora they contain)*;
checks.pl*	*subroutines to check that various variables are set, generating error messages otherwise* ;
subroutines*	*all major subroutines used throughout the system*;

Table 4: `/cgi-bin/w3c/lib:` Library scripts

4.1.1 Corpus Preparation

Suppose the sub-corpus is as given in Figure 11. This is *tokenized*, by the script `tokenize`, which yields two files: `xcorp.tok`, and `xcorp.item`.

`xcorp.tok` Tokenized corpus — the whole corpus, one word (i.e. word-token) or piece of punctuation per line (cf. the first column of Figure 13), and with paragraph breaks represented by `<P>` tags. Simply printing this file without the line breaks and with the `<P>` paragraph markers replaced by blank lines recreates the original corpus.

In what follows, we will call this the '`.tok` file'.

`xcorp.item` This file contains a character string, with one character per word-token, punctuation character, or blank line of the corpus (which means one character per line of the tokenized corpus file): the ith letter is `W` if the corresponding token in the `.tok` file is a word; `P` if it is punctuation, `S` if it is a paragraph separator `<P>`. Cf. Figure 12.

A glance at the first two columns of Figure 13 will show the relation between this file and the `.tok` file.

Given the information in this file it is possible to treat Words and Punctuation differently (e.g. when deciding how many items of context they want either side of a key word, users can decide whether to count punctuation). It is also possible to access the whole of the paragraph containing a particular hit (this is used in generating Context displays).

```
I believe he left his house to his friends, his money to the
poor, and his clothes to the nation.
```

Figure 11: Sample Corpus

```
WWWWWWWWWPWWWWWPWWWWWWWP
```

Figure 12: `xcorp.item`

The `.tok` file is further processed by the script `create_lexicon`, which yields 6 further files for each sub-corpus. These are summarized in Table 5.

Of these, `xcorp.lex` is a plain file, the others contain fixed length records (a detail we we will ignore in what follows).[31]

[31] That is, where we show the content of a file it will be unpacked, and will look like a plain text file.

xcorp.tok	xcorp.item	xcorp.seq
I	W	3
believe	W	5
he	W	8
left	W	11
his	W	9
house	W	10
to	W	16
his	W	9
friends	W	7
,	P	1
his	W	9
money	W	12
to	W	16
the	W	15
poor	W	14
,	P	1
and	W	4
his	W	9
clothes	W	6
to	W	16
the	W	15
nation	W	13
.	P	2

Figure 13: The tokenized corpus (`xcorp.tok`), its analysis as words, punctuation, etc. (`xcorp.item`), and the sequence file (`xcorp.seq`)

`xcorp.lex` This file contains a sorted list of the word-types that occur in the `.tok` file, given in lexicographic order (Figure 14), one type per line. One might think of this as the *lexicon* of the corpus. Looking at this file will answer the question "Does word X appear in this corpus?" (more generally "Does a word matching the pattern X appear in this subcorpus?"). On the other hand, if one thinks of the corpus as consisting of n word-types, $w_1, w_2, \ldots, w_n$, then this file allows one to answer the question "What is the string representation of word w_i" — e.g. in this case w_5 is `believe`, w_3 is `and`, and w_1 is the comma. See the first two columns of Figure 14.

`xcorp.seq` This contains a representation of the original corpus, but with word-tokens replaced by numbers (the type numbers given in `xcorp.lex`). Cf. Figure 13. To recreate the original corpus, one would look at each element w_i of the `.seq` file and print the string associated with w_i in the `.lex` file.

`xcorp.tok`	tokenized corpus: one word or punctuation element per line.
`xcorp.item`	records, for each token, whether it is a word or punctuation, or markup.
`xcorp.lex`	"Lexicon" — word-types (as strings) in lexicographic order: for each i the string associated with w_i .
`xcorp.seq`	the original corpus, but with i in place of the corresponding string.
`xcorp.lex.freq`	Frequency information for each word-type w_i.
`xcorp.lex.idx`	An index into `.lex`: where in `.lex` does the string associated with w_i occur?
`xcorp.lex.pos`	Lists of occurrences: the positions in `.seq` where tokens of w_i occur.
`xcorp.lex.pos.idx`	An index into `.lex.pos`: where in `.lex.pos` does the list of occurrences for w_i appear?

Table 5: Main Corpora Files

`xcorp.lex.freq` A representation of the word-type frequencies; each element is an integer — the ith element gives the frequency with which word-type w_i occurs in the corpus.

`xcorp.lex.idx` An index into the lexicon (`.lex`). This file records, for each word-type $w_1, \ldots, w_n$, at what record of the lexicon the string representation of w_i appears. For example, the relation between `xcorp.lex` and `xcorp.lex.idx` might be as in Figure 14: w_5 (*believe*) begins at record 10 in `xcorp.lex`

To find the printed representation of w_i, one retrieves the ith element of `.lex.idx`, and consults the specified record of the lexicon. This allows for rapid access to the string representation of any word in the corpus.

`xcorp.lex.pos` For each word w_i, the ith record of this file gives the positions (as byte off-sets) in the `.seq` where this word occurs (i.e. where in the corpus one can find the corresponding tokens).

`xcorp.lex.pos.idx` This gives an index into `.lex.pos`. It gives, for each word-type w_i, the record of `.lex.pos` at which the list of occurrences of w_i can be found.

The relationship between these files can be seen in Figure 14. To take a concrete example, w_{15} is the word "the". If we look at the 15th record of `.lex.idx`, we find the figure 70, this is the position in the `.lex` file where the string representation of this word begins. The string representation (`the`) can be found by opening `xcorp.lex` at position 70 (and reading forward the length of `the`). It occurs in the corpus 2 times, as witness the figure 2 in the 15th record of the `lex.freq` file. To find out where in the corpus it occurs, we look at the 15th record of the occurrences index — `.pos.idx`, which gives a value of 72. Opening the occurrences files `.lex.pos` at position 72, and reading the next 2 records (its frequency in the corpus) will tells us that "the" appears at positions 14 and 21 in the original corpus.

w_i	xcorp.lex	xcorp.lex.idx	xcorp.lex.pos	xcorp.lex.pos.idx	xcorp.lex.fr
1	,	0	10 16	0	2
2	.	2	23	8	1
3	I	4	1	12	1
4	and	6	17	16	1
5	believe	10	2	20	1
6	clothes	18	19	24	1
7	friends	26	9	28	1
8	he	34	3	32	1
9	his	37	5 8 11 18	36	4
10	house	41	6	52	1
11	left	47	4	56	1
12	money	52	12	60	1
13	nation	58	22	64	1
14	poor	65	15	68	1
15	the	70	14 21	72	2
16	to	74	7 13 20	80	3

Figure 14: Files giving Lexicon, Occurrences, Frequencies, and Indices

It will be seen from this that frequency information can be very rapidly calculated. In the case of a pattern matching a single word, one simply looks through the `.lex` file and for each line l where the pattern matches, and recovers record l from the `.lex.freq` file. Summing these gives the total number of hits in the corpus.

We will look at this in more detail in the following section.

4.1.2 Searching

When a user indicates that they want to use the search engine, the top level script (`w3c`) sets up a number of files (see below) and calls `bin/search`, which sets up the displays, and which in turn calls `bin/locate` which does the actual corpus search. Before these can operate, the user must first specify the corpus (and sub-corpus) they want to search, and what they want to search for (forms for doing this are generated by the scripts in `input`, see Table 6).

Suppose, for example, the user specifies the regular expression `[Nn]ice` and chooses the following sub-corpora of the Gutenberg corpus:

`drmnt10`	*10,000 Dreams Interpreted [Or . . . What's In A Dream]*	by Gustavus Hindman Miller.
`ahero10`	*A Hero of Our Time*	by M.Y. Lermontov.
`achoe10`	*A Child's History of England*	by Charles Dickens.
`cblad10`	*A Collection of Ballads*	by Andrew Lang.

`cgi-bin/w3c/input/`		
	`corpus`	*generate form for user to choose corpus ;*
	`subcorpus`	*generate form for user to choose sub-corpus;*
	`string`	*generate form for the user to select the type of search, and the search string;*
	`return`	*delete all search results from the current session, and return to the top w3c page;*

Table 6: Scripts used to set corpus, search type, and search string.

`325073`	*# Basic Search parameters*
`325073.options`	*# Other User options*

Table 7: Files generated for each search session.

Confirming these choices causes the files listed in Table 7 to be generated (the base file name `325073` uniquely identifies the session: it is arbitrary, generated when a user accesses the initial search page, or requests a new search).

These files define the environment for the current search, and for any subsequent search history (In the file listings below, comments follow the "#", and are added here for readability. In no case are they part of the actual file generated by the system.)

`325073` The 'base file' contains the data that controls the search. It is created when the user chooses the search-type, search-string, and sub-corpora to search, and it records these choices. The file is read by the `search` script, and updated if a user changes any of these during the current session. The numbers assigned to the sub-corpora are used in the other files. Thus, the entries here provide a mapping from the corpus numbers to the names of files which contain the actual corpora. The contents of this file, with explanatory comments, are in Figure 15.

`325073.options` This provides a record of the users search and display options; changing the search options changes the contents of this file. This file is created with default values by the `search` script. Part of this file, with comments, can be seen in Figure 16.

Confirming the choices also starts the search process, which is carried out by the `locate` script as follows.

First, the user's input is read from the base file, and the search term is split into 'components' — in general, a search term may consist of several components, each of which is intended to match a separate word. For example, a search term like `takes advantage` might be intended to find instances of the word *advantage* immediately following the word *takes*. Such a term consists of two components.

```
Type:Regular            # What kind of search is this?
Corpus:Gutenberg        # Which corpus is being searched?
String:[Nn]ice          # What is the search string?
Entire:                 # Is it over the whole corpus? (No)
Subcorpus:ahero10 3     # Which sub-corpora should be searched?
Subcorpus:achoe10 1     # Sub-corpus[3] is ahero10, cf. the .sub file.
Subcorpus:drmnt10 0     #
Subcorpus:cblad10 2     #
```

Figure 15: Contents of a base file: `325073`

```
Corpus:Gutenberg                    # Which corpus?
Sample:0                            # Where in the search history is this? (0=at the start)
Display_Initial:F                   # Which results should be displayed initially? (Frequency)
Display:10                          # How many items in the KWIC display at one time?
KWIC_L:6                            # How many words to the left of target in KWIC?
KWIC_R:6                            # How many words to the right of target in KWIC?
KWIC_Display_Reference:0            # Should the KWIC display show the sub-corpus's reference?
KWIC_Display_Delete:0               # Should there be a DELETE option in the KWIC display?
KWIC_Display_Position:0
KWIC_Display_All:0
KWIC_Display_Order:A
KWIC_P:0
CF_POS:0
CF:M
KWIC_Display_Range:1
Frequency_Lexical_Percentage:0
Collocations_General_Initial:SE
KWIC_Display_Position_Next:4
CF_M:0
Frequency_Individual_Order:MF
Frequency_General_Notes:1
CF_P:1
Frequency_Lexical_Notes:1
Sample_Size:0
Search_General_Initial:N
Frequency_Individual_Text:0
```

Figure 16: Part of a `.options` file, with comments.

325073.0.freq
325073.0.li
325073.0.results
325073.0.sub

Table 8: Files containing search results.

Next, the search string is normalized to a regular expression (e.g. the string `nice` becomes the pattern `^nice$` if the search type requested is 'exact match'). In outline, processing then proceeds as follows (this description is simplified in various ways, e.g. it does not address sorting of results).

+ For each sub-corpus selected:
  + for each component of the search pattern:
    + open the appropriate `.lex`, `.lex.freq`, `.lex.pos.idx`, and `.lex.pos` files; for the corpus;
    + read through the `.lex` file (which contains the corpus *types*) looking for matches. If a line matches this component of the search pattern, then:
      + if this is the first component of the search pattern, access frequency information and find the position information for the corresponding tokens, and store this information;
      + if this is not the first component, then store the position information in a number of arrays, $\texttt{looking}_1,\ldots,\texttt{looking}_n$, where $\texttt{looking}_i\texttt{[n]}$ is defined just in case the ith component of the search pattern matched at position n in the corpus.

      If the search term had only one component, then the start and end information for the position at which a hit was found is the same. If there were several components, then a hit has occurred only if the first component matched at position $n-1$, and looking$_1$[n] is defined, meaning that the second component matched at the adjacent position (and so on for any other components of the search pattern). The hit starts at the position of the first component, and ends at the position of the last.
    + This information about positions is recorded, as is a frequency information;
  + Frequency information is written out to the `.freq` file for this session. Position information is written in the `.results` file for this session.
+ For each sub-corpus searched, we recover the lexical entries (i.e. the strings) corresponding to the hits. To do this, we look in the `.seq` file for the corpus (where the corpus is represented as 'running text' but with word-type identifiers rather than strings), at the appropriate positions, and for each such position recover the identifiers of the word-types; looking in the `.lex.idx` file for the corpus in the appropriate place gives the location of the required string in the corpus's `.lex` file. String information is written onto the `.li` file for this session.

Reference	Subcorpus	Frequency
drmnt10	10,000 Dreams Interpreted [Or . . . What's In A Dream] by Gustavus Hindman Miller.	2
ahero10	A Hero of Our Time by M.Y. Lermontov.	1
achoe10	A Child's History of England by Charles Dickens.	1
cblad10	A Collection of Ballads by Andrew Lang.	0
	Total	4

Figure 17: Frequency results returned for search for `[Nn]ice` over indicated texts

achoe10	Holy Land, and afterwards died at	Venice	of a broken heart. Faster and
ahero10	so very elegant, his complexion so	nice	and white, his uniform so brand
drmnt10	dream of having an abundance of	nice	, clean crockery, denotes that you will
drmnt10	young woman to dream of a	nice	, ready-made shirt-waist, denotes that she will

Figure 18: KWIC results for returned for search for `[Nn]ice`

Certain other aspects of the process may be clarified if we consider the files produced by the search above. Results for this search, in terms of frequency and as a KWIC display are given in Figure 17 and Figure 18.

In addition to the two files mentioned above, this search produces the four files listed in Table 8. The `.0` in the names of these files indicates that they are the result of the initial search in this session. If the user 'refines' the search to produce a search history, then files with names `325073.1.freq`, `325073.2.freq`, etc. would be produced.

`325073.0.freq` Frequency results: a plain file pairing sub-corpora (identified by number) with numbers of occurrences. See Figure 22. The mapping of numbers to actual sub-corpus files is given in the base file.

`325073.0.li` A plain file containing the search results as strings. See Figure 21.

`325073.0.results` A collection of pairs of fixed length records, each record indicates a position in the lexicon of a sub-corpus, indicating the start and end positions of a sequence that matches the search term. Where the search string is just one component long, these are the same. See Figure 20. Which sub-corpus the match comes from is indicated in the `.sub` file.

`325073.0.sub` Records the association of hits to sub-corpora. Specifically, ith pair of records in the `.results` file occurs in the sub-corpus specified in the ith record of the `.sub` file. See Figure 19.

The relationship between all these files may be clarified by looking at Table 23.

```
1       # the first hit is from sub-corpus 1
3       # the second hit is from sub-corpus 3
0       # the third hit is from sub-corpus 0
0       # the fourth hit is from sub-corpus 0
```

Figure 19: A `.sub` file (unpacked): which sub-corpus contains each match

```
80892 80892       # the entry at achoe10 record 80892 corresponds to the string "Venice"
3150 3150         # ahero10 at 3150 corresponds to "nice"
51445 51445       # drmnt10 at 51445 corresponds to "nice"
165263 165263     # drmnt10 at 165263 corresponds to "nice"
```

Figure 20: A `.results` file (unpacked): the positions in the actual (the `.seq` file) of the hits.

```
Venice
nice
nice
nice
```

Figure 21: A `.li` file: the corpus strings matched by the search string

```
0:2       # Sub-corpus 0 is drmnt10, 2 hits
1:1       # Sub-corpus 1 is achoe10, 1 hit
2:0       # Sub-corpus 2 is cblad10, 0 hits
3:1       # Sub-corpus 3 is ahero10, 1 hit
All:4     # Total: 4 hits
```

Figure 22: A `.freq` file: how often the search string appears in each sub-corpus.

.results (Start-End Positions in Corpus)	.sub Subcorpus number for this session	Name of Sub-Corpus (from Subcorpus Array defined in the base file)	.li (the string that was matched)
80892 80892	1	achoe10	Venice
3150 3150	3	ahero10	nice
51445 51445	0	drmnt10	nice
165263 165263	0	drmnt10	nice

Figure 23: The relation between some of the files holding search results.

Given these results, frequency information can now be displayed immediately. If a KWIC display is requested for (say) the first result with (say) six words of context, then all that is required is to open the `.seq` file of the appropriate corpus (`achoe10.seq`) at record 80892, read six records either side, and for each record recover the associated string. Accessing wider context is similar, except that one should read backwards and forwards for the nearest paragraph markers (where these are is quickly recovered from the appropriate `.item` file).

4.2 Implementation of the Interface

The implementation of the interface is relatively unremarkable. It is based on html forms, which as already noted are generated by Perl programs. Frames are used extensively. The way the interface looks is determined by the scripts that generate the forms, and options which the user sets (recorded in the `.options` file for the session, as noted above). The scripts that allow users to set options interactively listed in Table 10, page 44.

The scripts that implement the interface are listed in Table 9, page 43.

4.3 Other Utilities

In this Section we will briefly document some of the other utilities contained in subdirectories of the top level directory:

`w3c/comments/` contains two scripts for dealing with users' comments. The comments themselves are held in a subdirectory.

`panel` *generates the form for users to input comments, submitting the form calls `process`;*

`process` *deals with the comments — essentially, it outputs to file a version of the form with the users input filled in;*

`replies/` *contains files of users comments;*

w3c/registration contains a single script `process` for dealing with users' requests for registration: it checks for completeness of information, and mails the information supplied by the user to the corpus administrator.

w3c/help contains two scripts: `help`, and `help-panel.` The former deals with users help request by returning an appropriate help page, according to the users location when help was requested (at its core is an array relating numbers generated by help requests with messages; the help messages themselves are in `/htdocs/w3c/help/`). The latter returns the user to their previous location after receiving help.

4.4 Discussion

Both the interface, and the programs used for searching are based on well-understood ideas, and are now very stable. The range of queries that can be handled is not as extensive as some other search engines offer (e.g. there is no support for investigating collocations, or deriving any but the most simple statistics). Nevertheless, the queries that can be handled are sufficient for a wide range of interesting questions to be addressed, and are probably sufficient for the intended users of the system. The interface is quite flexible in the ways it can be configured by the individual user, and results are presented in what we consider a satisfyingly clear way.

It should perhaps be admitted that that devising an acceptable interface to the corpora was rather harder than we had initially expected. There are at least three reasons for this:

- It is very easy to design simple forms in HTML to get input from web browsers, and implementing programs to perform searches on corpora is fairly straightforward (e.g. in a language like Perl). It is also easy to provide a link between these using cgi-scripts. The problem is that one needs more than just a simple interface. Standard HTML does not provide for interactive forms (i.e. forms which change their appearance interactively as the user makes choices), so one must present the user with a sequence of separate forms; these cannot be pre-prepared, but have to be composed by scripts (e.g. in Perl). These scripts are not complicated, but they are often large, and there are a large number of them, even for simple things. This makes maintenance and revision very time consuming.

- Web browsers are not at all standard in the way they interpret html. Of course this is not a surprise. What is surprising is how quirky and unpredictable the behaviour is: in testing two browsers (e.g. Netscape and Explorer) on two architectures (e.g. Windows and Unix) it was quite common for us to find four different kinds of behaviour, all undocumented, of course. Particular problems in designing forms arise from variability in fonts, which means that what looks fine under one set up can be incomprehensible under another. There is no alternative to exhaustive testing, and trial and error. This is very time frustrating and time consuming.

- There are no good tools for debugging html-form/cgi-script/Perl script combinations. Again, inspired trial and error is the only possibility.

5 Information Pages

5.1 Description

The "information" pages are intended to give access to general information about Corpora and Corpus Linguistics, to provide a general overview of the field and to help users find other Web resources and printed material that may be useful. Some of the information provided is introductory (intended for those with no background), some is general reference information, which will be useful to anyone involved in Corpus Linguistics. The information provided is generally quite detailed — the level of detail and discussion are comparable to what one will find in some of the introductory text books that have appeared over the last year or so, for example. The information has been carefully structured and categorized. An approximation of the content of the top level information page can be seen in Figure 24.

The pages provide the following:

Introduction: some general questions:

What is a Corpus?

What is Corpus Linguistics?

Background A history of Corpus Linguistics, from the earliest times to the present.

Using Corpora Some of the issues you may need to think about if you intend to use a Corpus.

Choosing a Corpus Issues that arise in choosing a Corpus.

Corpus Compilation Some hints about how to set about compiling a Corpus.

Corpus Annotation A brief discussion of the issues that arise in annotating Corpora for various different purposes. This is a topic that is covered well by other web sites, so the discussion here is quite brief, supplemented with links to the other sites, especially to the pages supplementing the book *Corpus Linguistics* by Tony McEnery and Andrew Wilson McEnery and Wilson (1996). [32]

Research areas Brief discussions of the many areas where Corpus study can be useful (including Computational Linguistics, Cultural Studies, Discourse Analysis and Pragmatics, Grammar/Syntax, Historical Linguistics, Language Acquisition, Language Teaching, Language Variation, Lexicography, Linguistics, Machine Translation, Natural Language Processing, Psycholinguistics, Semantics, Social Psychology, Sociolinguistics, Speech, and Stylistics).

List of Corpora A list of over 100 Corpora, listed alphabetically, but also classified in various ways, with a short description, and links indicating where more information is available.

Glossary A glossary of some of the technical terms used in the pages, and links to two major sources of such information: *Corpus Linguistics* by Tony McEnery and Andrew Wilson McEnery and Wilson (1996), and the *Systematic Dictionary of Corpus Linguistics*.[33]

Bibliography A list of key publications, in the field, together with links to major sources of bibliographic information.

Related sites A list of 50 or so major corpus-related WWW sites, classified under several headings (including General Sites, Project Sites, Research centres/groups, Text centres, Mailing Lists, Journals) .

Software Lists of sites offering descriptions of software, or actual software for Corpus related activity: General Sites, sites offering software relating to Concordancing, and Tagging and Parsing.

Courses Information about on-line courses, and conferences relating to Corpus Linguistics.

Tutorial A link to the W3Corpora Tutorial (see Section 6).

These pages represent a valuable resource for anyone starting out in Corpus Linguistics, and much information of value to established researchers. In terms of level of detail, comprehensiveness, systematicity and clarity, they bear comparison with anything currently available.

5.2 Directory structure

The information pages are to be found in the directory `/w3c/corpus_ling/`.[34] The directory structure for these pages is as follows can be seen in Table 11, and Table 12.

6 Tutorial

6.1 Description

The tutorial first of all goes over the basic procedure of using the search engine to search a corpus, and then goes on to discuss in detail some of the tasks a user might want to perform.

The idea is to describe, with practical illustration, some of the areas where corpora are useful. Each section describes a problem/research question and shows how it can be addressed using the W3Corpora tool. As well as the practicalities of carrying out the search, there is discussion of how the results can be interpreted. Each section is concluded with suggestions for further exercises. The tasks described are:

[32] `http://www.ling.lancs.ac.uk/monkey/ihe/linguistics/contents.htm`
[33] `http://donelaitis.vdu.lt/publikacijos/SDoCL.htm.`
[34] The web address is `http://clwww.essex.ac.uk/w3c/corpus_ling/`.

W3-Corpora project. Comments and suggestions are welcome on email or through our Comments Page

Corpus Linguistics

These pages have been created as part of the W3-Corpora Project at the University of Essex.

Introduction

Welcome to the W3C Corpus Linguistics pages!

Here you can read about corpus linguistics and find many interesting links to other sites. Use the buttons to the left to access the different pages or follow the links in each section. You can also use the 'back' and 'forward' buttons on your browser (e.g Netscape, Internet Explorer, etc.) to navigate between the pages.
These pages are to a large extent based on material available on the Internet. Use this site for a general overview of the field and for finding the pages that are of interest and use to you. Remember that you can use the on-site Search Engine to make your own corpus searches.
Contents

NEXT | Search Engine

Figure 24: Information Pages: top level.

- look for the meaning of a word: this is illustrated by an exploration of what can be found about the meaning of *annotation.*
- compare two similar words/synonyms: illustrated by exploration of the difference between *sick* and *ill.*
- compare how a word is used in different kinds of text (how are words like *love, kinetic, man, he,* and *education* used in different kind of text?)
- see which preposition to use: illustrated by considering the question "Which preposition do I use after 'explanation' in a sentence like 'I want to find an explanation . . . this'?"
- check the spelling of a word: to look up the spelling of a word in a dictionary is not easy (you have to know how it is spelled. . .). However, the possibility of a regular-expression search over a Corpus means one can find out about spelling even if only relatively little is known. In case there are variants, one can choose the variant that is most common (or most common within a particular text type).
- starting-point for further explorations: how to play with a corpus — who knows what interesting lines of inquiry a user may find?

6.2 Directory Structure

These pages are to be found in the directory `w3c/help/intro/`.[35] The directory structure can be seen in Table 13.

7 Corpora

7.1 Description

Currently, the W3Corpora site makes three corpus collections available:[36]

Gutenberg The Gutenberg Project is a collection of texts, many of them whole books, in the public domain. The bulk of the texts were written in the 19th century or earlier. Only a subset of all Gutenberg texts are available through the W3Corpora search engine (new texts are added to the Gutenberg collection almost daily). However, this still gives access to 321 texts, totally around 19,000,000 words.

[35] The web address is `http://clwww.essex.ac.uk/w3c/help/intro/`.

[36] It cannot be said often enough that we are grateful to the people who collected these corpora, and who administer them.

LOB The London-Oslo Bergen Corpus (LOB). This consists of about 1 million words of written British English from 1961. The 500 texts are classified into 15 different text categories. Access to this requires registration.

Tagged LOB This is the tagged version of LOB Corpus, where words have been assigned a part-of-speech tag. Access to this requires registration.

The access restrictions are a reflection of the agreement under which the corpora are made available to us (and to all other users), and are necessary because the corpora contain copyright material.

These are relatively small corpora by modern standards (though the Gutenberg collection is of respectable size), and do not provide the sort of up-to-date view of English that some other corpora provide. On the other hand, they are widely used and well regarded 'standard' collections. In addition, the site allows users to upload and search their own corpora, simply by FTP-ing the text to the web server (detailed instructions are provided). Over the next few months, we hope to install corpora from the On-line books initiative (which are freely available, without copyright).

In addition, an interesting feature of the site is that fact that is possible for users to up-load and search their own corpora, so one might add a fourth corpus:

User Defined Corpora It is possible for users to use the search tool to search any text they like, simply by FTP-ing the text to the web server (detailed instructions are provided at the W3Corpora site).

7.2 Directory Structure

The directory structure of the corpus pages is summarized in Table 14, page 48. For discussion of the functions of the various files, see Section 4.

8 Conclusion and Evaluation

In the main, the W3Corpora site and search engine satisfy the desiderata that were listed at the outset. It is usable without the need to install or download programs, to register or get authorization. The interface is relatively 'friendly', and searches return a useful minimum of information (frequency and KWIC, with access to wider context on demand). It is possible to perform some operations on the results of searches (e.g. 'refinement'). It is possible to install and search user defined corpora. The search engine is supported by extensive help and general documentation (as readers may judge for themselves by visiting the site). One may reasonable claim that this is a good place to start learning about corpus linguistics.

The number of users making corpus searches appears to be stable, at about 100 per month, the number of users accessing the information pages is more than ten times that (these figures are

do not take into account access from within the local, `essex.ac.uk`, domain). These are not very impressive figures by Internet standards, of course, but the number of people using the information pages is gratifying, and one must bear in mind that the intended audience for the search pages is not very large (one would expect that anyone who develops a serious interest in corpus linguistics would look for a local installation of corpus manipulation tools with more specialized, or simply larger, corpora). The goals of the project are satisfied if people are able to get a feeling for whether corpus linguistics has anything to offer them without excessive effort.

The W3Corpora site and search engine has been used and evaluated by a number of people: experts (members of the project Consultative Committee)[37], novice (student) users, who have used the pages on their own and in a class-room/teaching context, and others who have accessed the pages directly without supervision. The evaluation process involved requesting users to give feed-back, either by filling out a questionnaire (available from the site) or via e-mail.[38]

In general, the response from the users of the site has been positive. The site is said to be 'very useful' (primarily student users) or 'fairly useful' (expert and advanced users) and easy to use. Negative criticisms have been raised in relation to the interface to the search engine interface ('too much clicking', 'frames mess up display'), and to the limited corpus resources. Users reported that they did not use the help texts to any greater extent, which would seem to indicate that the search engine is self-instructing and easy to use. In general, users with access to more advanced concordancing programs found the tool too restricted in scope. Generally less advanced users (students and other users) found the tool more useful than the advanced/expert users. Most users, irrespective of background and access to other resources, were positive about the general idea of the project and the site.

The site has been used as a tool for teaching in three different environments at the Universities of Essex, Uppsala, and Verona. The teachers involved all report that they found the site useful, especially for use with novice users. The students were positive and found the site easy to use. Observations in the classroom indicate that some students were able to use the site without any help from the teacher while some users needed guidance (particularly ones with little previous experience of using computers and/or web-based resources).

Despite the generally positive evaluation, there are both minor and major limitations that should be pointed out. It is not really possible to use *any* web browser: providing results in a readable format requires the use of HTML frames, and some (obsolete) browsers do not do this. The operations that can be applied to search results are limited to 'refinement' of searches: to do anything more, the user has to use standard tricks (e.g. 'drag and drop') to save results to their own machine. The range of searches that can be carried out is not very exciting: there are no tools for investigating collocations, for example. There is no provision for languages other than English. Perhaps most seriously, the fact that only public domain corpora can be made freely available means that only limited and rather unsystematic corpora are available.

It is worth considering this last point at greater length. The corpus resources available at the W3Corpora site are sufficient for some interesting work, and certainly enough for a typical novice

[37] See (Arnold et al., 1999, Appendix A) for a list.

[38] The questionnaire can be accessed at `http://clwww.essex.ac.uk/w3c/corpus_ling/about.html`.

user to cut their teeth on. However, they are not as extensive, rich or varied as one would like, nor are they all freely and immediately available without registration. It is worth saying a word about why this is the case.

The project was never intended as a Corpus collection exercise. Corpus collection is very difficult at a purely practical level, and involves theoretical questions that we did not feel able to address (what *sort* of Corpus is one collecting? is it intended to be balanced or representative in any way?). It is also a long-term activity, in which we felt little progress could be made in the life of this project.

Moreover, there have been a number of important corpus collection exercises in recent years, at least some of which have produced results that are widely and freely available for research and educational purposes. The project was launched on the assumption that it was foolish to replicate this work, and that it would be possible to re-use these results. Though reasonable, this assumption turns out to have been false. The effect of this is that the number of corpora, and their range and variety of coverage is not as great as we had originally hoped.

It is very easy to confuse 'widely and freely available' with 'public domain': the latter means that anyone can do what they like with a resource, the former that the author or owner of the resource retains a degree of control. We were not confused about this, but we were wrong about the status of even the most widely distributed and used corpora. Corpora which we thought were in the public domain turned out to be just widely and freely available. This meant there was no problem in obtaining them, but we needed to seek permission to make them accessible over the web. This too, we had foreseen, and the proposal even contained an allocation of resources to obtain legal opinion if necessary. In the event, such opinion would be otiose.

We had reasoned that in making available a corpus over the WWW one is actually allowing access to only a very small part of any particular document. For example, a search might return every instance of the word *nice* with five words of context on either side. Even with a very common word (like *the*), one would not be getting very much of the content of the document, and certainly not in any form that one could easily use for any purpose other than linguistic study. Thus, we reasoned that allowing such access might reasonably fall under the kind of "fair use" clause of copyright that allows one to quote from published sources without obtaining copyright clearance. It turns out that the correctness or otherwise of the reasoning here is irrelevant, because of the sociology of corpus collection.

When one approaches a corpus administrator with a request such as ours, one typically receives a warm statement of support for one's general aims, continuing something like this: "… However, in collecting this corpus, I and others have approached thousands of copyright holders and obtained their permission to use relevant material in very restricted ways; the kind of access you are proposing may go beyond this (anyway, I cannot be sure that the copyright holders will not see it as going beyond this), so I would have to approach all of these thousands of copyright holders again, and seek further permission.… This I am unwilling to do."

This is, of course, a perfectly reasonable and understandable reaction. In particular, it is a reaction that one cannot argue with: it is quite pointless to argue that what one is proposing falls under the spirit of the copyright agreement, or that no actual infringement can actually occur. For it is

not enough to convince the corpus administrator of this, one must convince him or her that any copyright owner would also be so convinced, and that all this is so obvious that there is no need to even consult the copyright owner. This is impossible.

It must be admitted that the Corpus resources that are immediately available at the W3Corpora site fall short of what one would like. In the best case, there would be access to standard Linguistic corpora including various kinds of annotation, available without registration. It turns out this is impossible, because the standard Linguistic corpora contain material which is under copyright, and are not in the public domain. For public domain material, one has to turn to collections like the Gutenberg collection, and the On-Line Books Initiative, which lack the systematicity of standard Linguistic corpora.

As the WWW matures, as better tools become available for developing web applications, it should be become easier to develop applications that improve on W3Corpora in many ways. Unfortunately, the lack of tagged public domain corpora is a social, not technological, phenomenon, so in this respect the situation is unlikely to improve greatly in the near future. Really getting started in corpus linguistics is going to require some kind of registration to use a non-public domain corpus, and hence a significant commitment on the part of the user.

References

Bas Aarts, Justin Buckley, and Gerald Nelson. Internet Grammar of English, 1999. `http://www.ucl.ac.uk/english-usage/internet-grammar/`.

D.J. Arnold. WWW-IGE: World Wide Web access to Corpora and the Internet Grammar of English. In *Proceedings of DRH-97 (Digital Resources in the Humanities)*, pages 711–716, St. Anne's College Oxford, Sept 1997.

D.J. Arnold. World wide web access to corpora. *Cuadernos de Filologia Inglesa de la Universidad de Murcia*, 9(1):125–145, 2000. Pascual Cantos Gomez, editor.

Doug Arnold. Web access to corpora: the W3Corpora project. In *Post-Conference Workshop on Computer and Internet Supported Education in Language and Speech Technology*, page ***, University of Bergen, Norway, June 1999. European Association for Computational Linguistics.

Doug Arnold, Bas Aarts, Justin Buckley, Ylva Berglund, Gerald Nelson, and Martin Rondell. Corpora and grammars on the web: the W3Corpora–IGE Project, final report JTAP–2/247. `http://clwww.essex.ac.uk/w3c-ige/FinalReport/`, February 1999.

Doug Arnold and Ylva Berglund. WWW access to corpora: a tool for teaching and learning about corpora. In *TALC-98 (Third International Conference on Teaching and Language Corpora)*, Keeble College, Oxford, 24–27 July 1998. Humanities Computing Unit, Oxford University, Oxford. `http://clwww.essex.ac.uk/w3c/corpus_ling/TALC.html`.

Natalia Brines-Moya and Julie Hartill. Criteria for user-oriented evaluation of monolingual text corpora interfaces. In *Proceedings of the First International Conference on Language Resources and Evaluation*, volume 2, pages 893–898, Granada, Spain, 28-30 May 1998.

Tony McEnery and Andrew Wilson. *Corpus Linguistics*. Edinburgh University Press, Edinburgh, 1996.

`/w3c/display/`	*Scripts controlling the display:*
`panel`	*Script generating main panel of function buttons: [HELP]/[DISPLAY]/[FREQUENCY]/[SEARCH]/[OPTIONS].*
`match/`	*Scripts relating to display of results in "KWIC" form, and associated "Context":*
`frame/`	*Sets up the frame layout for "KWIC" and "Context" pages.*
`kwic/`	*Scripts relating to the display of search results in KWIC form:*
`frame*`	*Sets up the frame layout for "KWIC" pages*
`panel*`	*Generates "KWIC FRAME" and associated buttons (for scrolling through KWIC listings)*
`panel_random*`	*Generates "Random Sample" button;*
`display*`	*Generates and displays KWIC results from the corpus files generated by the initial corpus search: for example, tells the user how far through the KWIC listing she is; sorts the results, if necessary;*
`key/`	*Users can opt to see which subcorpus KWIC results come from; if selected, information about the subcorpus appears in the "KEY" frame:*
`frame*`	*Sets up frame layout.*
`panel*`	*Generates "KEY" heading.*
`display*`	*Displays Corpus names in the "KEY" frame, in response to requests.*
`context/`	*Scripts relating to the display of wider "Context" (i.e. outside KWIC range)*
`frame*`	*Sets up frame layout*
`panel*`	*Generates "Context Frame"*
`display*`	*Finds and displays, on request, the context (e.g. paragraph) surrounding a particular hit;*
`frequency/`	
`frame*`	*Sets up frame layout*
`panel*`	*"Frequency Screen", and associated buttons: [MATCH FREQUENCY]/[SUBCORPUS FREQUENCY]/[LEXICAL FREQUENCY]*
`match*`	*Displays "Match Frequencies", i.e. overall frequency (total number of hits)*
`lexical*`	*Displays "Lexical Frequencies", i.e. frequency associated with each hit*
`individual*`	*Displays "Individual Frequencies", i.e. number of hits from each subcorpus*
`search/`	
`frame*`	*Sets up frame layout*
`panel*`	*"Search Screen" and buttons: [NEW SEARCH]/[REFINE SEARCH]/[SEARCH HISTORY]*
`new*`	*Script for starting a new search;*
`refine*`	*Generates input to get parameters for a "refined" search;*
`history*`	*Generates a table showing the history of current search (e.g. original search plus any refinements)*
`sample*`	*Performs a "refined search" on request*
`time*`	*Updates the "sample number" (referenced in the search history)*
`options/`	*Scripts to generate html forms allowing users to set user definable options. See Table 10.*

Table 9: `/w3c/display/`: Scripts controlling the display.

`frame*`	*Sets up the frame layout for the main option pages.*
`panel*`	*Generates the panel of function buttons for the main Options Screen: [General Options]/[Display Options]/[Frequency Options]/[Search Options].*
`general*`	*Generates the form allowing user to set General Options (e.g. what screen should be displayed initially).*
`match/`	*Scripts for setting "Display Options" (e.g. relating to KWIC, and how matching is carried out):*
`frame*`	*Sets up frame layout.*
`panel*`	*Generates "Display Options Screen" panel of function buttons: [GENERAL OPTIONS]/[KWIC OPTIONS]/[CONTEXT OPTIONS].*
`general*`	*Generates form for setting "General Match Options".*
`kwic*`	*Generates form for setting "KWIC Match Options" — how KWIC results are displayed.*
`context*`	*Generates form for setting "Context Match Options" — options relating to how context is displayed in KWIC search results (e.g. number of words left/right).*
`frequency/`	*Scripts for setting options relating to the display of results showing frequency:*
`frame*`	*Sets up frame layout.*
`panel*`	*Generates "Frequency Options Screen" panel of function buttons: [GENERAL OPTIONS]/[MATCH OPTIONS]/[INDIVIDUAL OPTIONS]/[LEXICAL OPTIONS]*
`general*`	*Generates form for "Displaying General Frequency Options" (e.g. what is the initial Frequency Screen?)*
`match*`	*Generates form for "Displaying Match Frequency Options"*
`individual*`	*Generates form for "Displaying Individual Frequency Options" — i.e. those relating to individual sub-corpora (e.g. Display order: e.g. most frequent first, least frequent first?)*
`lexical*`	*Generates form for "Displaying Lexical Frequencies Options" — how lexical frequencies will be displayed*
`search/`	*Scripts for setting options relating to searches:*
`frame*`	*Sets up frame layout.*
`panel*`	*Generates "Search Options Screen" panel of buttons: [GENERAL OPTIONS]/[NEW SEARCH OPTIONS]/[REFINE SEARCH OPTIONS]/[SEARCH HISTORY OPTIONS]*
`general*`	*Generates form for "Displaying General Search Options" (What should the initial search screen be?)*
`new*`	*Generates form for "Displaying New Search Options" (none at present).*
`refine*`	*Generates form for "Displaying Refine Search Options" (none at present).*
`history*`	*Generates form for "Displaying Search History Options" (none at present).*
`new*`	*Script for actually updating user selected display options (writes users selected options out to the* `.options` *file).*

Table 10: `/w3c/display/options/`: Scripts to generate html forms allowing users to set user definable options.

about.html	*General Information about the W3-Corpora Project*
questionnaire.html	*The Evaluation Questionnaire that was sent out*
bibliography/bibliography.html	*Bibliography*
content/	
index.html	*The top level Information page*
introduction.html	*Introduction: main window of* `index.html`
introduction2.html	*What is a Corpus?*
introduction3.html	*What is Corpus Linguistics?*
history.html	*Background*
introduction4.html	*Using Corpora*
using.html	*Choosing a Corpus*
compilation.html	*Corpus Compilation*
using2.html	*How can a Corpus be used?*
search_engine.html	*Access to the search engine*
corpora/types/annotated.html	*Corpus Annotation*
research/research.html	*Research Areas*
corpora/list/index2.html	*List of Corpora*
glossary.html	*Glossary*
sites/sites.html	*Related Sites*
software.html	*Software*
courses/conferences.html	*Courses*
examples.html	*Accessed from* `glossary.html`
corpora/	*See Table 12*

Table 11: Directory Structure of Information Pages: `w3c/corpus_ling`

```
content/corpora/
            list/          Lists of Corpora:
                  index.html
                  index2.html
                  public/        Publicly available Corpora:
                        ECI.html
                        susanne.html
                        gutenberg.html
                        childes.html
                  private/       Corpora requiring a License:
                        brown/
                              brown.html
                              brown_list.html
                        LOB/
                              lob.html
                              lob_list.html
                              lobtagged.html
                              lobuntagex.html
                        bnc.html
                        llc.html
                        lancaster.html
                        air.html
                        kolhapur.html
                        longman.html
                        helsinki.html
                        market.html
            types/         A dicussion of different kinds of Corpus:
                  annotated.html
                  parallel.html
                  reference.html
                  comparable.html
            samples/       Samples of some Corpora:
                  susanne.html
                  gutenberg.html
                  childes.html
                  brown.html
                  bnc.html
```

Table 12: Directory Structure of Information Pages: `w3c/corpus_ling/content/`

`start_page.html`	*Welcome to the tutorial for the W3-Corpora Interface. Description of the search procedure for when using the W3-Corpora search engine. List of suggested uses.*
`compare_educat.html`	*Example: W3-Corpora as a starting-point for further explorations — comprehensive introduction to corpus linguistic methodology.*
`compare_educat_education.html`	*Illustration:* education
`meaning.html`	*Example: look for the meaning of a word. How to use the search engine to look for the meaning of a word.*
`meaning_annotation.html`	*Illustration: "What does* annotation *mean?"*
`meaning_mammal.html`	*Illustration: "What sort of word is* mammal*?"*
`synonyms.html`	*Example: Studying synonyms,*
`synonyms_ill.html`	*Illustration:* sick *and* ill
`compare.html`	*Example: Compare how often a word occurs in different kinds of texts.*
`compare_love.html`	*Illustration:* love.
`compare_kinetic.html`	*Illustration:* kinetic.
`compare_man.html`	*Illustration:* man.
`compare_he.html`	*Illustration:* he.
`compare_education.html`	*Illustration:* education.
`preposition.html`	*Example: Which preposition?*
`preposition_explanation.html`	*Illustration: "I want to find an explanation . . . this".*
`preposition_explanation2.html`	*Discussion: "I want to find an explanation . . . this".*
`spelling.html`	*Example: Check spelling.*
`spelling_colour2.html`	*Illustration:* color *or* colour*?*

Table 13: Directory Structure of Tutorial Pages: `w3c/help/intro/`.

```
/w3c/corpus/
      Gutenberg/     Texts from the Gutenberg Corpus
            achoe10        Text of 'A Child's History of England by Charles Dickens.'
            achoe10.item
            achoe10.word.lex.freq
            achoe10.word.lex
            achoe10.word.lex.idx
            achoe10.word.lex.pos
            achoe10.word.lex.pos.idx
            achoe10.word.seq
            ...
            markup.list    Information on how these texts are marked up into paragraphs
      User/          User defined texts
            xcorp.item
            xcorp.word.seq
            xcorp.word.lex.freq
            xcorp.word.lex.pos
            xcorp.word.lex
            xcorp.word.lex.idx
            xcorp.word.lex.pos.idx
            ...
            markup.list    Information on how these texts are marked up into paragraphs
      Lob/           Texts from the London Oslo Bergen Corpus
            A              The LOB corpus is divided into section A...R
            A.item
            A.word.lex.freq
            A.word.lex
            A.word.lex.idx
            A.word.lex.pos
            A.word.lex.pos.idx
            A.word.seq
            ...
            markup.list
```

Table 14: Corpus Directories (truncated).