Terminology Extraction:
Legacy Terminology in Content Authoring and Translation

Hanne Smaadahl[1] and Frederik Fouvry[2]

Abstract

In this paper, we discuss various challenges in extracting and processing legacy terminology found in existing documentation and incorporate these into authoring, translation and terminology processes. This paper describes the extraction, evaluation, and processing of term candidates for use in an end-to-end global translation management system (GTMS).

1. Introduction

When Business Objects first began using an end-to-end global translation management system (GTMS) and terminology management module, one of the challenges we faced was how to successfully incorporate source terminology found in existing documentation into the new authoring, translation and terminology tools and processes.

Our authoring processes include the acrocheck™ tool from acrolinx,[3] a quality assurance tool which can be integrated into the editing processes used by our technical writers. Acrocheck uses a combination of style rules and terminology to guide the writers and improve the overall quality of the source text.

The terminology management module of our GTMS is a concept-based, multilingual database. The terminology database contains key terminology with target language equivalents for up to 11 languages.

The purpose of this project was two-fold. First, we needed to capture all existing technical vocabulary for use in our authoring environment. Second, we wanted to document new technical terminology for use with our global translation management system (GTMS).

Given the high volume of legacy content, any solution would have to rely on some level of automation. In this paper, we will discuss how the legacy content was processed for terminology, analyze the terminology output, and describe how the output was incorporated into the authoring and translation processes.

This paper is organized as follows: Section 2 provides a brief overview of what is considered a term in the context of Business Objects. Section 3 describes the corpus that was used to extract the terminology. Section 4 details the preparation and procedure for extracting terms from this corpus. An analysis of the terminology extraction output is presented in section 5. Section 6 outlines how the output was

[1] Senior Terminologist, Business Objects
New: hsmaadahl@businessobjects.com Hanne.Smaadahl@businessobjects.com
[2] Senior Linguistic Engineer, acrolinx
e-mail: Frederik.Fouvry@acrolinx.com
[3] See http://www.acrolinx.com/.

processed and applied in the GTMS and authoring environment, while section 7 provides conclusions and some thoughts on future work with the data.

2. What is a Term?

The currency of a terminology management system is the actual term entries it contains. As defined in ISO 12620:1999(E) a term is a "designation of a defined concept in a special language by a linguistic expression" (p. 5). Terminological units (terms) are typically used in specialized discourse by subject matter experts by whom they have been acquired through a learning process (Cabré Castellví, 2003, p. 185). A term may consist of one or more words, may contain symbols and can have variants, e.g. different forms of spelling (for example a full form and an abbreviation). The precise meaning of terms is context dependent. Though terms may coincide with words in general language, it is their context, or use in a specialized subject field that determine their meaning.

For the purpose of terminology management, it is useful to categorize terms according to vocabulary type (general vs. technical) and the time when they came into use (new terms vs. old terms). Figure 1 illustrates how terms can be divided into four quadrants based on these two factors: existing general vocabulary, new general vocabulary, existing technical vocabulary, and new technical vocabulary.

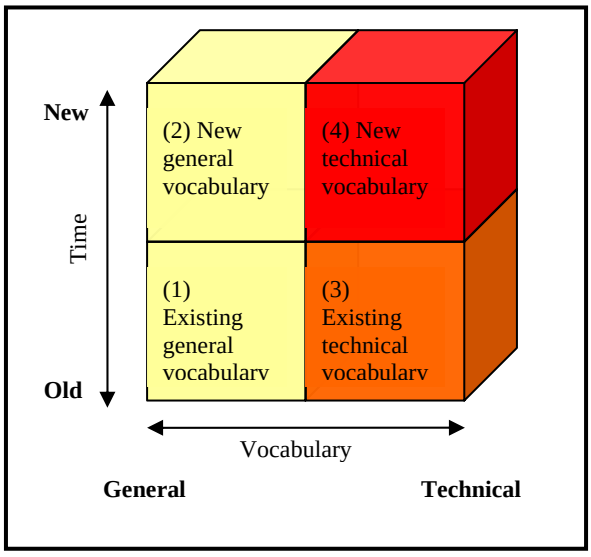


Figure 1: The four terminology quadrants.

A typical terminology management system will focus on technical terminology (quadrants 3 and 4). As the system matures, terminology management efforts will be centred on new technical terminology (quadrant 4).

For Business Objects, we have identified a list of 14 term categories that we consider to fall within quadrants 3 and 4 (see Table 1 below). To qualify as a term, a verbal designation must belong to one of these categories. However, it is important to note that just because a word or a phrase falls within any of these categories does not mean that it is automatically documented in our terminology database. Our focus is on "added value": we document the terms that are high visibility, importance, difficulty, etc. There will always be an element of subjectivity and judgment.

Term category	**Description**
base form	A term whose inflected forms are also part of the approved, corporate terminology, such as common nouns, technical terms and any terms that designate a technical or key concept in Business Objects' products or offerings.
back-end component	Name of functionality that is not accessed by the end-user, but must run in the background for front-end functionality to be available.
documentation title	Name of a documentation title.
feature	Name of functionality found inside a product and which is accessible only when the product is running.
file name	Name of a file.
front-end component	Name of functionality that can run on its own even when the product is not running, e.g. components launched off the Start menu in Windows-based applications.
other proper name	A term that denotes a single, specific object.
Product	Name of a full product.
Product line	Name of a set of integrated products that are marketed as a cohesive solution to specific BI challenges.
service/solution	Name of a solution or service offering such as support, training, consulting, or bundling.
standardText	A fixed chunk of recurring text including slogans, tag lines and names of campaigns or campaign themes.
technology	Name of a specific technology or group of technologies, including protocols, scripting techniques and services (but excluding Business Objects' products, components and features).
user interface	Any chunk of text that is part of a software application and is visible to the end-user (on menus and toolbars, in dialogs, etc.). For the terminology database, the following UI labels will be documented: menu name, menu command, command button label, tab name, toolbar button label, and web navigational element.
web page title	The text that appears in the title bar of a window opened via the user's web browser.

Table 1: Term categories at Business Objects.

3. The Corpus

The corpus consists of software documentation and users' guides in a collection of XML files. The XML files are in the Darwin Information Typing Architecture (DITA) format, an XML-based architecture for authoring technical documentation (see DITA, 2005).

DITA makes use of two file types: dita files and ditamap files. To create a document in DITA, topics are authored in individual XML files, and then organized in a hierarchical sequence known as a DITA map. The DITA map resembles a table of contents and organizes references to DITA topics for compilation into deliverables like PDF, online help and CHM files (Microsoft Compiled HTML Help). Multiple DITA maps can reuse the same topics to produce different deliverables. It goes beyond the scope of this paper to describe the DITA architecture in greater detail. Please refer to http://dita.xml.org/ for more information on the DITA standard. A sample DITA XML document from the corpus is shown in Appendix A.

During the extraction, only XML and DITA files were processed. The corpus consisted of 37283 XML files; the total word count for the corpus was 3581466 words in 823092 sentences.

4. Data Preparation and Term Extraction Procedure

4.1 What is acrocheck™ and what does it do?

The term extraction was performed by acrolinx. We used acrocheck and the acrocheck Batch Client to collect the term candidates. Acrocheck is a quality assurance tool for technical documentation in English, German, French, etc. It is typically used in the following scenario: text is sent from an editing tool (such as Microsoft Word, XMetaL, etc.) to the acrocheck server, which processes the text looking for spelling errors, violations of grammar rules and (customizable) style rules and for the use of deprecated terminology. The results are returned to the writer along with correction suggestions. A sample is shown in Figure 2.

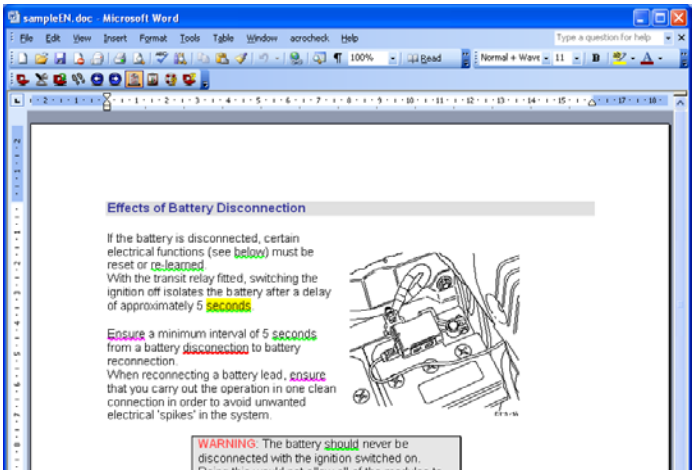


Figure 2: A screen shot of a document checked by acrocheck™.

Acrocheck tokenizes the incoming text and provides it with linguistic annotations such as token classes, part-of-speech tags and morphological analyses. The checks are performed with rules and algorithms that combine these sources of information.

Acrocheck can also process XML input. In that case, the markup can support the interpretation of the tag content. In XHTML[4] for instance:

- A “p” or “h1” element indicates the beginning and/or end of a sentence.
- The “em” element does not indicate a sentence break at all: it just occurs within the text flow and can be ignored, but its content should be processed.
- The “acronym” element indicates a token and its content should be considered as one unit.
- For linguistic processing “img” elements do not have any meaning, and can be skipped.

Some markup languages contain elements such as “indexterm” which – depending on its actual usage by the technical writers – may be useful for detecting phrases and term candidates.

The term extraction procedure mainly runs in the background, i.e., the acrocheck server collects the candidate terms from each submitted piece of text, but they are normally not displayed to the user. It builds on the same linguistic information as the checking components. The term extraction results are exported from the system in the OLIF[5] format.

4.2 Preparation

For the project that we describe in this paper, a large number of files had to be processed. We also wanted to use the language processing capabilities of acrocheck to detect potential new terms. With the acrocheck Batch Client (see Figure 3), the user can select files, configure the processing, run and send them in one batch to the acrocheck server.

[4] The “XML-version” of HTML. See http://www.w3.org/TR/xhtml1/.
[5] The Open Lexicon Interchange Format. See http://www.olif.net/.

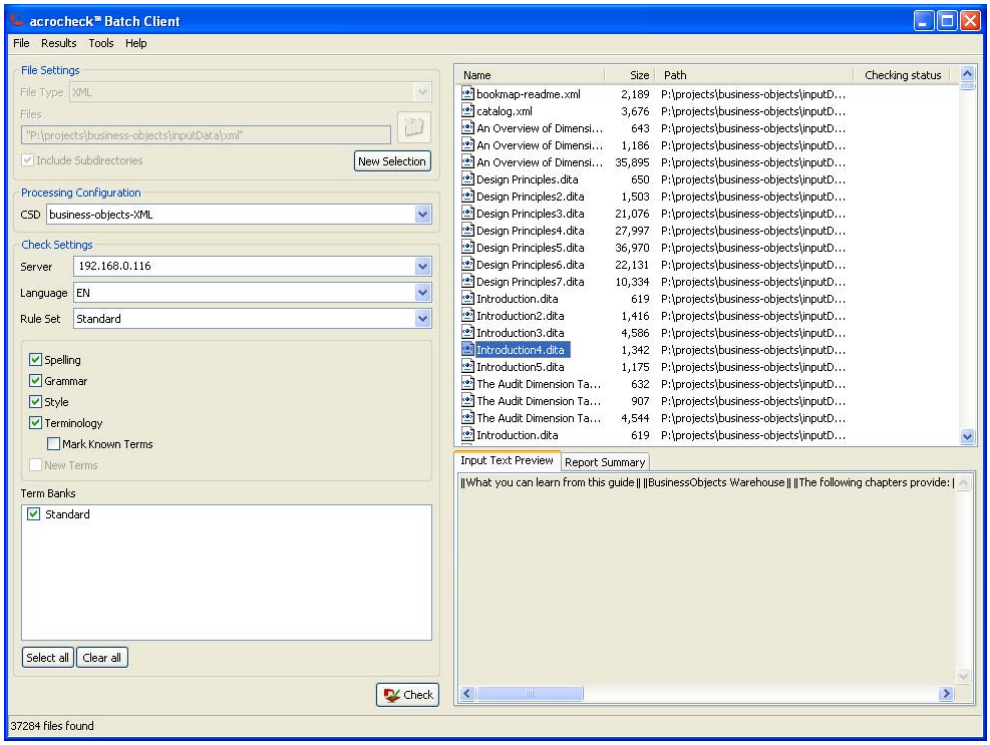


Figure 3: The acrocheck™ Batch Client ready to run a check.

The texts were first prepared for term extraction by making sure that all XML is well-formed and can be processed, by adding obvious missing vocabulary, retraining the part of speech tagger and configuring the server for DITA XML processing (setting skipped elements, included elements, sentence break elements and so on). The next step was to run a subset of the data collection through the term extraction procedure, and submit the results for inspection to Business Objects. The results are stored in a machine-readable format with fields for the lemmatized term, its frequency, the contexts it is found in, the files of origin, and part-of-speech information. This information is available to help with the validation of the term candidates. Validation consists of assigning a status (e.g. approved, deprecated, non-term, preferred, proposed, and so on) to a term candidate, thereby taking it up into the terminology.

Using feedback from the first round, we adapted the term extraction rule configuration and then all files were put through term extraction. For the delivery of the results, the term collection was split into two groups: one for terms that only occurred once (terms with frequency 1), and one group of terms with a frequency of at least 2. The reason for the separation is that terms of frequency 1 are less likely to be terms, especially in larger corpora.

4.3 Extraction Technique

The terminology extraction uses linguistic pattern matching to detect term candidates. For instance, in an English text a noun preceded by a noun and followed by a noun is a term candidate. The pattern for this example might look as follows:

```
POS:NN      POS:NN      POS:NN      45
```

Finite verbs are only rarely parts of terms. To exclude them from terms, the pattern could look like this (a third person singular verb surrounded by anything):

```
0     POS:VBS     0     -100
```

Acrocheck allows matching on several linguistic properties: string, token class, part-of-speech tag, morphology (including compound analysis), and whether a word already is (part of) a term. If a pattern matches, the token gets a score (45 in the example). The scores of all matching patterns are added up, and if the result is higher than a given threshold, the token is considered to be a term candidate. Candidates are then automatically filtered against additional criteria to reduce the number of false positives. The criteria we used include:

- Sequences of term candidates are concatenated into one, larger term candidate.
- Terms must not contain any stop words.
- Term candidates must be unique.
- Words which are unlikely to be terms on their own, but may form one in combination with other words (e.g. “window”, “button”) are treated specially: they are not extracted, except when they are part of a larger term.
- The server can be configured to include (suspected) spelling errors in the set of term candidates.
- Existing terms are not extracted again.

The extraction patterns are manually specified, using human linguistic knowledge, and refined using the results of term extractions from technical documentation. The term extraction component of acrocheck contains a set of standard patterns for each language, which can be adapted to the need of specific customers and text types.

4.4 Word Count

In the following paragraphs, we will be referring to the number of words in the documents. The word count is determined by the tokenization: in acrocheck a token is counted as a word. In most cases, a word corresponds to the usual notion of “word”, i.e., a group of characters surrounded by spaces and/or punctuation. In some cases, words have been taken together, e.g. with dates or abbreviations: the entire date is taken together as one word. That is also the case for abbreviations.

5. The Output

The term extraction resulted in 72157 term candidates. A term candidate in this context corresponds to unique types in the corpus; a type may consist of one or more words. The frequencies range from 1 to 25054, with 24590 term candidates with a frequency of 1 and one term candidate with a frequency of 25054. The median frequency was 286 and average frequency was 623.

The output was tabularized with each occurring frequency and the number of unique term candidates or types at each frequency. Each frequency was given a ranking from 1, the highest frequency (= 25054), to 532, the lowest frequency (= 1). Appendix B shows each rank, frequency, types (number of unique terms) and tokens[6] per frequency.

As shown in Table 2, almost 90% of all term candidates (89.2%) can be found in frequency 1 through 10. Term candidates with a frequency of 1 and 2 makes up 60% of all term candidates (60.7%) with 34% of all term candidates at frequency 1 and almost 27% at frequency 2.

Rank	Frequency	# of term candidates	% of total # of term candidates
532	1	24590	34.1%
531	2	19204	26.6%
530	3	5157	7.1%
529	4	6527	9.0%
528	5	1685	2.3%
527	6	2967	4.1%
526	7	935	1.3%
525	8	1620	2.2%
524	9	785	1.1%
523	10	908	1.3%
		64410	**89.2%**

Table 2: Number of term candidates for frequencies 1 through 10, and percentages of total number of types (unique term candidates).

The number of term candidates with a rank greater than 100 (frequencies from 1 through 743) makes up over 99 percent of all term candidates (99.9% or 72054 types). Similarly, 99 percent of all term candidates have a frequency below the *median* (frequency lower than 286) and below the *average* (frequency lower than 623), the percentages being 99.6% and 99.8%, respectively.

Of the term candidates with a rank between 1 and 100, the top 59 term candidates ranked by frequency have only one type each. Only 3 of the top 100 frequencies have more than one term candidate (in each case 2), which makes the total number of unique term candidates for the top 100 frequencies a mere 103 or 0.001% of the total number of unique term candidates. The actual term candidates for the top 10 frequencies are shown in Table 3 with their frequency, rank and a sample context.

[6] Tokens in this context are the total number of terms, not number of individual words.

Term	Rank	Frequency	Sample context
Report	1	25054	This feature displays your report in HTML format instead of true RPT format; you can return to the Design tab to make adjustments for the best results when viewing the report over the web.
User	2	10088	You can create a web-based report designer that allows the user to modify reports with the RAS server.
Object	3	9399	When you schedule an object that has been published in multiple languages, you can generate instances of the report in one or more languages.
Example	4	9205	For more information on this example and other dashboard-related resources, please see Resources.
BusinessObjects Enterprise	5	8215	When you're not connected to BusinessObjects Enterprise, you can use the Crystal Reports Offline Viewer to look at Crystal reports that you've downloaded from BusinessObjects Enterprise.
Report	6	8125	The SecurityInfo property of the Report has an ObjectPrincipals collection that contains a list of all users and groups with rights to the Report.
formula	7	4740	Variables make complex formulas easier to decipher because they break the formulas up into manageable components.
BusinessObjects Enterprise SDK	8	4527	It provides a detailed set of lessons that teach you how to develop a web application using the BusinessObjects Enterprise SDK and a number of tutorials that teach you how to use the BusinessObjects Enterprise SDK to perform both client and administrative tasks.
name	9	4293	Unless your objects are very precisely named, then a restriction may not be obvious to the user simply from the name of the object.
group	10	4257	You can calculate the standard deviation for all values within a group (for example, sales grouped by the state that they come from).

Table 3: Top ten most frequent terms identified by term extraction.

5.1 Comparing Results against a General Corpus

In a general corpus, very high frequency words are typically function words or closed class words. For example, in the British National Corpus (BNC) which contains 100 million words, the most frequent word is *the* which occurs 6187267 times and accounts for just over 6% of the corpus (6.2%). The second-most frequent word, *be*, occurs 4239632 times (slightly over 4% at 4.2%), followed by *of* which occurs 3093444 times (just over 3%) (Kilgarriff, 2006). Only 117 vocabulary items are needed to account for half the British National Corpus.

Rank	Frequency	Term	Part of speech
1	6187267	the	det
2	4239632	be	v
3	3093444	of	prep
4	2687863	and	conj
5	2186369	a	det
6	1924315	in	prep
7	1620850	to	infinitive-marker
8	1375636	have	v
9	1090186	it	pron
10	1039323	to	prep
Total	**25444885**		

Table 4: Top 10 most frequent words in the BNC (Kilgarriff, 2006).

According to Kilgarriff's frequency list, all top 10 most frequent words in the BNC are function words (grammatical words like determiners, prepositions, conjunctions, pronouns, the infinitive marker and auxiliary verbs). These types of words are of little interest in a terminology management system that documents and organizes concepts in a subject matter area. When looking for concepts, we should instead focus on content words or lexical words (nouns, verbs, adjectives, and some adverbs).

The highest ranking noun in the BNC is "time" with a rank of 53 and a frequency of 183427. Among the 117 most frequent words in the BNC (accounting for 50% of the corpus), only seven are nouns, whereas the lower frequency words in BNC are almost exclusively content words (nouns, verbs, adjectives, and adverbs).

Since our term extraction method excluded all closed word classes, all the high frequency words found in a general corpus such as the BNC are filtered out through stop word lists, leaving only the content words which would normally have the lower frequencies in a general corpus. The term candidates with the highest frequencies seem to be relatively polysemous with an "open" meaning defined primarily by their contexts. In that sense, the term extraction results are similar to the BNC with the types increasing in their denotative value with decreasing frequency.

As illustrated in Figures 4 and 5 below, when we look at individual tokens the term extraction data still follows a distribution akin to a general corpus. Figures 4 and 5 illustrate a simple 1/f function similar to Zipf's law applied to the BNC corpus and the term extraction results, respectively (Zipf, 1949). For the BNC corpus, the calculations include the words ranked 1 through 117, which accounts for 50% of the word count in the BNC corpus. The term extraction data covers all types with a frequency above the median (286), or the top 321 types. In both figures, the relative frequency numbers are plotted alongside the 1/f function calculated on the frequency numbers. As can be seen from these figures, the distributions in both the BNC and term extraction results – unsurprisingly – follow Zipf's law.

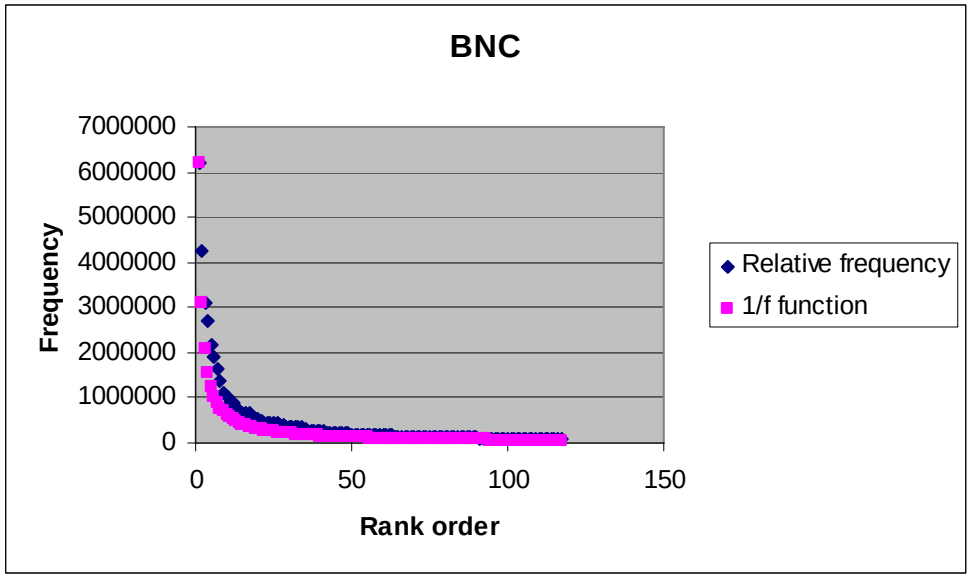


Figure 4: Zipf's law applied to the BNC corpus.

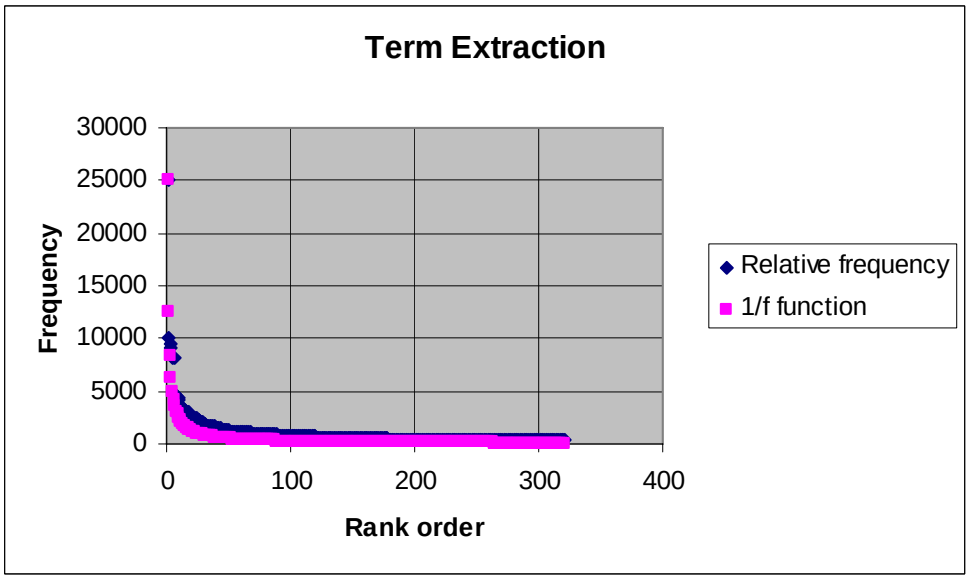


Figure 5: Zipf's law applied to the term extraction results.

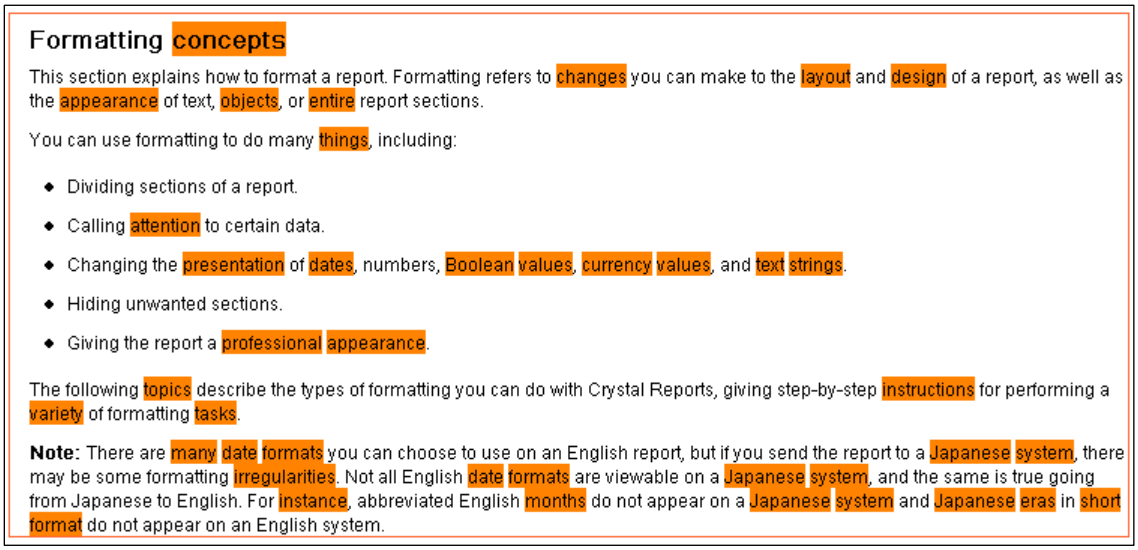
6. Application

The purpose of this project was two-fold. The first requirement was to capture all existing technical vocabulary for use in our authoring environment. Second, we wanted to document new technical terminology for use with our global translation management system (GTMS).

6.1. Application in Authoring Environment

For the authoring environment, it was decided to treat all the term candidates that resulted from the term extraction as legacy, or existing terms, and add these to a term bank that would act as a de facto stop word list for the legacy terms. The result being that whenever the system encountered a term that had been extracted, it would ignore the term for the purpose of term acquisition. Given the high volume of existing content, a relatively new terminology management system, and limited resources to process new terminology, this approach allows us to move our terminology management strategy directly into quadrant 4 (see Figure 1) at less cost and time compared to a more traditional approach of manual terminology extraction.

Figure 6 shows the result of running the terminology extraction in acrocheck on a sample DITA file prior to the term extraction project. New terms are marked in orange. In this sample, a lot of terms are marked as new. For authors to process and submit all of these potential terms into a terminology management system would add considerable overhead to their work.



Formatting concepts

This section explains how to format a report. Formatting refers to changes you can make to the layout and design of a report, as well as the appearance of text, objects, or entire report sections.

You can use formatting to do many things, including:

- Dividing sections of a report.
- Calling attention to certain data.
- Changing the presentation of dates, numbers, Boolean values, currency values, and text strings.
- Hiding unwanted sections.
- Giving the report a professional appearance.

The following topics describe the types of formatting you can do with Crystal Reports, giving step-by-step instructions for performing a variety of formatting tasks.

Note: There are many date formats you can choose to use on an English report, but if you send the report to a Japanese system, there may be some formatting irregularities. Not all English date formats are viewable on a Japanese system, and the same is true going from Japanese to English. For instance, abbreviated English months do not appear on a Japanese system and Japanese eras in short format do not appear on an English system.

Figure 6: Result of terminology extraction in acrocheck on a sample DITA file.

When the term checker is run on the same file after term extraction, it yields the results shown in Figure 7. Compared to the pre-term extraction result, the number of terms that would need review is considerably reduced. A review of the potential term candidates shows that this sample file would yield only one potential term for further processing: *Japanese system*.

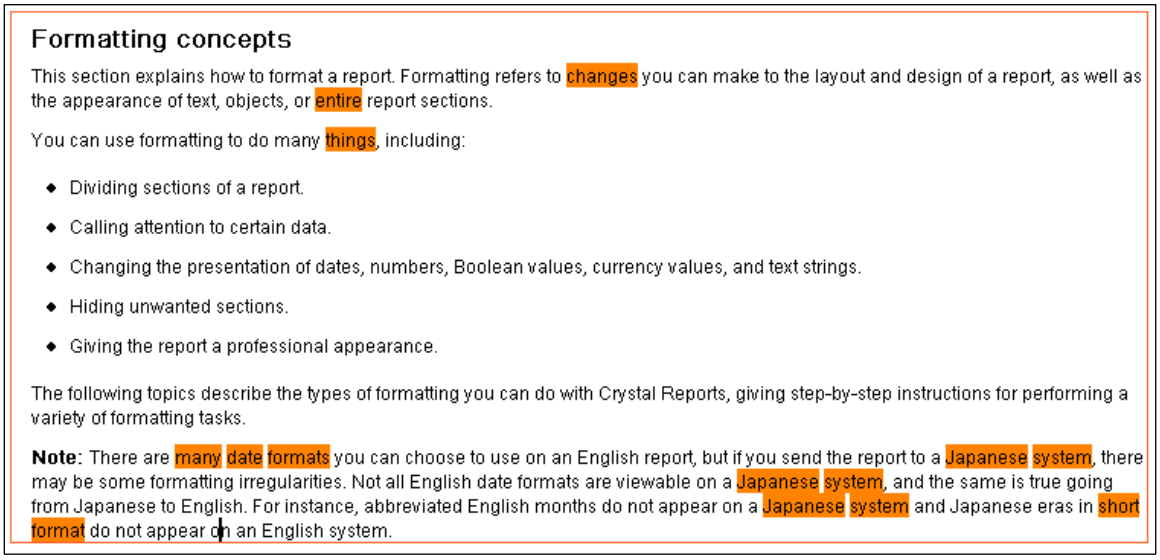


Figure 7: Result of terminology extraction in acrocheck on a sample DITA file after the output of term extraction has been applied.

6.2. Application in Global Translation Management System

For the GTMS, we were looking to develop criteria to assign priority to the individual term candidates for manual processing. Given the high volume of term candidates and the limited resources available to process them, it is critical that the list of term candidates be narrowed down as much as possible based on objective criteria prior to any manual processing. The criteria used at the time this paper was written were based on frequency, an aspect of morphology, and a set of non-term indicators. We hope to further improve and refine these criteria in the future.

6.2.1. Term Candidates by Frequency

A term candidate is more likely to be a term the higher its frequency. There are several explanations for this phenomenon. First, the goal of a technical document is to convey information in an unambiguous and precise manner. Thus, the nature of technical writing demands a high level of structure, consistency, repetition and reuse of technical concepts, which in part explains why technical terms occur with such high frequencies.

Second, our terminology management system is based on an assumption of return on investment (ROI) whereby the cost of each terminological entry goes down as its use increases. In other words, terminological entries that are leveraged often offer a better ROI than those that are leveraged infrequently. In a writing environment where consistency is paramount, term candidates that occur only once have little interest as there would be no consistency at issue. The first decision was therefore to exclude all term candidates with a frequency of 1 from the manual processing efforts.

The top 1000 most frequent term candidates were manually reviewed by the terminologist and a group of technical writers/editors. Among the top 1000 most frequent terms, 661 term candidates were deemed to be non-terms whereas 339 term candidates were considered to be terms for further processing. The non-terms where discarded from the GTMS, but still included in the stop word list for the authoring

environment. The valid term candidates were prepared according to our terminology database standards, and imported into our terminology database. This in turn initiated tasks in our GTMS system for adding foreign language equivalents for each new term.

6.2.2 Term Candidates by Compound Pattern[7]

Terms behave differently depending on whether they are single-word terms or multi-word terms. Multi-word terms are more likely to be monosemous, whereas single-word terms are typically more polysemous and their meaning is commonly influenced by their context (Jacquemin, 2001, p. 9). Single-word terms are therefore more fluid in both their meaning and migration between disciplines and more likely to transition between general and technical domains (some examples from the domain of business intelligence include *cube*, *universe*, *dashboard*, and *report*).

Prior to the term extraction project, we had a number of existing terminology resources or collections. These included 944 terms in a relatively new terminology database as well as 2439 terms from various personal term lists. These were all hand-picked terms by language professionals (translators, language specialists, or terminologists) and the assumption is that they made it on to these lists because they were high visibility, importance, difficulty, etc. We used these 3383 terms as a baseline to determine what types of compounds where more likely term candidates. As shown in Table 5, 70% of terms from these lists are compounds consisting of two or more words. Based on this analysis of earlier word lists and our existing terminology database, we determined that term candidates that are multi-word terms are more likely to be terms compared to single word terms.

Compound pattern	Number	Percentage
1 word	1034	30.6%
2 words	1342	39.7%
3 words	572	16.9%
4 words	246	7.3%
5 words	99	2.9%
6 words	42	1.2%
7 words	25	0.7%
8 words	8	0.2%
9 words	5	0.2%
10 words	3	0.1%
11 words	3	0.1%
12 words	3	0.1%
16 words	1	0.0%
Total Number of terms	**3383**	**100.00%**

Table 5: Term count and percentage of terms for the various compound patterns (number of words in compound) that were found in legacy term lists.

[7] A "compound pattern" refers to the number of words that make up a particular term candidate. The patterns are restricted to individual words, not morphemes.

For the term extraction results, Table 6 shows the number of terms and percentage for each compound pattern for term candidates with a frequency greater than 1, equal to 1 and in total. As can be seen in this table, about 75% of all the term candidates consist of two or more words. For the term candidates that occurred only once, the number is almost 84%. For the term candidates with a frequency of two and higher, 72% of term candidates had a compound pattern of two or more words. This closely matches the patterns identified in the existing terminology lists.

Compound pattern	Frequency >1	% freq > 1	Frequency = 1	% freq = 1	Total	% total
1 word	13341	28.0%	4028	16.4%	17369	24.1%
2 words	20560	43.2%	10071	41.0%	30631	42.5%
3 words	9361	19.7%	6778	27.6%	16139	22.4%
4 words	2885	6.1%	2361	9.6%	5246	7.3%
5 words	1044	2.2%	968	3.9%	2012	2.8%
6 words	293	0.6%	288	1.2%	581	0.8%
7 words	64	0.1%	59	0.2%	123	0.2%
8 words	12	0.0%	22	0.1%	34	0.0%
9 words	5	0.0%	8	0.0%	13	0.0%
10 words	1	0.0%	3	0.0%	4	0.0%
11 words	0	0.0%	4	0.0%	4	0.0%
12 words	1	0.0%	0	0.0%	1	0.0%
Total # of terms	**47567**	**100.0%**	**24590**	**100.0%**	**72157**	**100.0%**

Table 6: Term count and percentage of terms for the various compound patterns (number of words in compound) that were found in term extraction results.

As discussed in the previous section, term candidates with a frequency of 1 are excluded from further processing. The top 1000 high frequency term candidates were included for processing regardless of compound pattern. Of the 661 term candidates that were deemed to be non-terms, 575 were single-word terms and 86 where multi-word terms. Of the 339 term candidates were considered real terms, 100 were single-word terms and 239 where multi-word terms. Table 7 shows the distribution of non-terms and terms for each compound pattern for the top 1000 most frequent types.

Compound patterns	Term candidates	Non-terms	Real terms	% non-terms
1 word	675	575	100	85.2
2 words	253	66	187	21.1
3 words	54	14	40	25.9
4 words	17	5	12	29.4
5 words	0	0	0	n/a
6 words	1	1	0	n/a
Total	**1000**	**661**	**339**	

Table 7: Distribution of non-terms and potential real terms for each compound pattern for the top 1000 most frequent types.

This confirms the tendency of terminological units to be multi-word terms, and the decision to exclude single-word terms from the manual processing beyond the top

most frequent types. By including the top 100 most frequent single-word terms, we still cover the most frequent terms that straddle the general and technical (business intelligence) domains.

A review of the compound patterns of 6 or more words revealed that most of these term candidates were sentence fragments rather than technical terms. The exception being names of documentation titles (for example: *BusinessObjects Enterprise XI R2 Portal Integration Kit User's Guide*, *BusinessObjects Enterprise XI R2 Portal Integration Kit Administrator's Guide*, and *BusinessObjects Enterprise XI R2 Portal Integration Kit Installation Guide*). Since documentation titles can be collected through other channels, all term candidates consisting of 6 or more words were excluded from manual processing.

After excluding single-word terms and terms consisting of 6 or more words, 33611 term candidates with compound patters of 2, 3, 4, and 5 words are left to be processed.[8]

6.2.3 Term Candidates by Non-Term Indicators

For the remaining 33611 term candidates that met the frequency and compound pattern criteria, we developed non-term indicators that we use to evaluate the candidates' suitability for inclusion in the terminology database. Any of the remaining 33611 term candidates that contain or begin or end with any of the indicators in Table 8 were flagged as non-terms.

Term candidate contains:	**Term candidate begins with:**	**Term candidate ends with:**
Numbers from 0-9	double letters: aa, bb, cc, …, zz	The string "adjust"
Ampersand &	Ordinals: first, second…,ninth	The string "appear"
Angle brackets < and >	The string "access"	The string "application"
At-sign @	The string "add" or "adding"	The string "close"
Backslash \	The string "appropriate"	The string "command"
Bar |	The string "available"	The string "consist"
Caret ^	The string "BusinessObjects"	The string "contain"
Colon :	The string "choose"	The string "control"
Curly brackets { and }	The string "close"	The string "file"
Double space	The string "format"	The string "folder"
Ellipsis …	The string "install"	The string "format"
Equal =	The string "integrate"	The string "group"
Forward slash /	The string "open"	The string "integrate"
Full stop .	The string "PLN"	The string "label"
Hash sign #	The string "print"	The string "method"
Parenthesis (and)	The string "save"	The string "now"
Percentage %	The string "schedule"	The string "open"
Plus +	The string "select"	The string "provide"
Semi colon ;	The string "set"	The string "provider"
Square brackets [and]	The string "show"	The string "query"
Underscore _	The string "tab"	The string "replace"
The string "allow"	The string "table"	The string "report"
The string "cannot"	The string "target"	The string "represent"
The string "check"	The string "this"	The string "server"

[8] This excludes the 239 multi-word terms that appeared among the top 1000 most frequent terms, as these have been processed already.

Term candidate contains:	Term candidate begins with:	Term candidate ends with:
The string “check box” or “checkbox”		The string “tab”
The string “delete”		
The string “dialog box”		
The string “edition”		
The string “expose”		
The string “feature”		
The string “get”[9]		
The string “guide”		
The string “hide”		
The string “list box”		
The string “map box”		
The string “menu”		
The string “model”		
The string “navigational box”		
The string “obtain”		
The string “print”		
The string “receive”		
The string “send”		
The string “specify”		
The string “text box” or “textbox”		
The string “title box”		
The string “tutorial”		
The string “value”		
The string “version”		
The string “window”		

Table 8: Non-term indicators.

Filtering the remaining 33611 term candidates according to these criteria resulted in 12350 types with non-term indicators and 21261 potential real terms. The distribution of non-terms (types that contains non-term indicators) and potential real terms for each compound pattern (2-word, 3-word, 4-word and 5-word compound patterns) is shown in Table 9. Based on the currently identified non-term indicators, the percentage of non-terms increases with the length of the compound pattern with 32% of 2-word types identified as non-terms and as many as 69% of 5-word types identified as non-terms.

Compound pattern	Term candidates	Non-terms	Potential terms	% non-terms
2 words	20382	6489	13893	31.8%
3 words	9315	3616	5699	38.8%
4 words	2870	1529	1341	53.3%
5 words	1044	716	328	68.6%
Total	**33611**	**12350**	**21261**	

Table 9: Distribution of non-terms and potential real terms for each compound pattern

[9] The string “get” ensures the exclusion of programming functions, methods and operations such as *GetPrograms method*, *GetCaption function*, and *getDocumentInformation operation.*

This increase in the percentage of non-terms with the longer multi-word term candidates is consistent with the findings from the top 1000 most frequent terms that were processed manually.

7. Conclusions and Future Direction

In this paper, we have described how term extraction can be applied to real-world content development and translation processes. In the case of Business Objects, a large corpus of existing product documentation was processed for term candidates. The resulting list of term candidates was then incorporated into our content authoring processes as part of the ongoing effort to identify and manage new technical terminology. This approach accelerated our terminology management efforts to a level that would usually be seen in more mature systems.

The number of term candidates for manual term processing was reduced from 72157 down to 21600[10] types, or by 70%, by excluding all types with a frequency of just one, any single-word types that did not appear among the top 1000 most frequent terms, all types consisting of 6 or more words, and all types matching any of the non-term indicators. Given the high number of potential terms still left to process, it is clear that we need further criteria to narrow down our efforts.

At the time of writing, the experience with the version of the term extraction component that we used for this paper has been used to extend and improve term extraction functionality. It is now using a more flexible pattern mechanism to filter more precisely, based on richer linguistic criteria, and by e.g. removing generic words. This reduces the number of undesirable term candidates, and with it the time that is needed to validate them. Another improvement is that information about the term candidates is added to the results: why is a term candidate being proposed? Filters such as the criteria that were mentioned in the previous section will be included in the new extraction rules.

Since the number of terms that is used in technical documentation is very large, validating the terms will always be a major task. However, more ways to flexibly filter and classify the term candidates will make it easier. The acrocheck terminology component for instance can work with term rules, which are term patterns, rather than terms themselves, e.g. “program” + Noun. This is one already existing way to classify and describe terms, in which not every term needs to be validated explicitly for checking. When instances of this pattern are found, they can be added automatically to the term base, because the pattern has been validated already. Complex filters over linguistic properties such as the reasons why a term candidate has been proposed, make it possible to retrieve a group of term candidates that can subsequently be validated as a group in one step. We are making the first experiences with the new setup, but the improvements are already making validation easier and more interesting.

[10] The 21261 potential multi-word terms plus the 339 terms from the top 1000 most frequent term candidates.

References

British National Corpus (BNC), http://www.natcorp.ox.ac.uk/.

Cabré Castellví, M. T. (2003) Theories of terminology: their description, prescription and explanation. *Terminology*, 9(2), 163–99.

DITA (2005), *OASIS Darwin Information Typing Architecture (DITA) Architectural Specification v1.0* , and *OASIS Darwin Information Typing Architecture (DITA) Language Specification v1.0*. For more general information about the standard, see http://www.oasis-open.org/committees/ tc_home.php?wg_abbrev=ditaand http://dita.xml.org/

ISO 12620 : 1999 (E). *Computer applications in terminology — Data categories.*

ISO 1087–1 : 2000. *Terminology work — Vocabulary — Part 1: Theory and application*.

Jacquemin, C. (2001). *Spotting and Discovering Terms through Natural Language Processing* (Cambridge, Massachusetts Institute of Technology: The MIT Press).

Kilgarriff, A. (2006). *BNC database and word frequency lists*. http://www.kilgarriff.co.uk/bnc-readme.html

Leech, G., Rayson, P. and Wilson, A. (2001). *Word Frequencies in Written and Spoken English* (Harlow: Longman). http://www.comp.lancs.ac.uk/ucrel/ bncfreq/flists.html.

Zipf, G. K. (1949). *Human Behavior and the Principle of Least Effort*. (Cambridge, MA: Addison-Wesley).

Appendix A: Sample DITA XML document

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE dita PUBLIC "-//OASIS//DTD DITA Composite//EN"
"dtd_1.0/ditabase.dtd">
<dita>
<concept xml:lang="en-us" id="fm_2006081551_117878">
<title>Formatting concepts</title>
<prolog>
      <critdates>
            <created date="2006-08-03"/>
      </critdates>
      <metadata>
            <audience type="other" othertype="Power User" job="other"
                  otherjob="Professional Report Creator"/>
            <keywords>
                  <indexterm>formatting</indexterm>
            </keywords>
            <prodinfo>
                  <prodname>Crystal Reports</prodname>
                  <vrmlist>
                      <vrm version="XI" release="R2"
                           modification="00000"/>
                  </vrmlist>
            </prodinfo>
      </metadata>
      </prolog>
<conbody>
<p>This section explains how to format a report. Formatting refers to
changes you can make to the layout and design of a report, as well as the
appearance of text, objects, or entire report sections.</p>
<p>You can use formatting to do many things, including:</p>
<ul>
    <li><p>Dividing sections of a report.</p></li>
    <li><p>Calling attention to certain data.</p></li>
    <li><p>Changing the presentation of dates, numbers, Boolean values,
currency values, and text strings.</p></li>
    <li><p>Hiding unwanted sections.</p></li>
    <li><p>Giving the report a professional appearance.</p></li>
</ul>
<p>The following topics describe the types of formatting you can do with
Crystal Reports, giving step-by-step instructions for performing a variety
of formatting tasks.</p>
<note type="note">There are many date formats you can choose to use on an
English report, but if you send the report to a Japanese system, there may
be some formatting irregularities. Not all English date formats are viewable
on a Japanese system, and the same is true going from Japanese to English.
For instance, abbreviated English months do not appear on a Japanese system
and Japanese eras in short format do not appear on an English system.</note>
</conbody>
</concept>


</dita>
```

Appendix B: Term frequencies

Table showing frequency rank, frequency, types (number of unique terms) and tokens per frequency (tokens in this context are the total number of terms, not individual words):

Rank	Freq.	Types	Tokens
1	25054	1	25054
2	10088	1	10088
3	9399	1	9399
4	9205	1	9205
5	8215	1	8215
6	8125	1	8125
7	4740	1	4740
8	4527	1	4527
9	4293	1	4293
10	4257	1	4257
11	3592	1	3592
12	3437	1	3437
13	3341	1	3341
14	3207	1	3207
15	3040	1	3040
16	3022	1	3022
17	2976	1	2976
18	2638	1	2638
19	2596	1	2596
20	2487	1	2487
21	2471	1	2471
22	2468	1	2468
23	2430	1	2430
24	2391	1	2391
25	2367	1	2367
26	2249	1	2249
27	2168	1	2168
28	2139	1	2139
29	2093	1	2093
30	1944	1	1944
31	1830	1	1830
32	1799	1	1799
33	1796	1	1796
34	1735	1	1735
35	1685	1	1685
36	1676	1	1676
37	1641	1	1641
38	1624	1	1624
39	1590	1	1590
40	1561	1	1561
41	1525	1	1525
42	1469	1	1469
43	1446	1	1446
44	1381	1	1381
45	1372	1	1372
46	1358	1	1358
47	1330	1	1330
48	1324	1	1324
49	1278	1	1278
50	1275	1	1275
51	1228	1	1228
52	1225	1	1225
53	1199	1	1199
54	1194	1	1194
55	1157	1	1157
56	1154	1	1154
57	1148	1	1148
58	1115	1	1115
59	1109	1	1109
60	1082	2	2164
61	1081	1	1081
62	1061	1	1061
63	1059	1	1059
64	1058	1	1058
65	1056	1	1056
66	1047	1	1047
67	1002	1	1002
68	1000	1	1000
69	996	1	996
70	975	1	975
71	973	1	973
72	946	1	946
73	945	1	945
74	933	1	933
75	929	1	929
76	919	1	919
77	901	1	901
78	900	1	900
79	897	2	1794
80	895	1	895
81	873	1	873
82	872	1	872
83	868	1	868
84	863	1	863
85	861	1	861
86	855	1	855
87	854	1	854
88	849	1	849
89	838	1	838
90	837	1	837
91	827	1	827
92	822	1	822
93	819	1	819
94	818	1	818
95	790	1	790
96	772	1	772
97	756	1	756
98	751	1	751
99	746	1	746
100	745	2	1490
101	743	1	743
102	722	2	1444
103	721	1	721
104	713	2	1426
105	707	1	707
106	703	1	703
107	700	1	700
108	699	1	699
109	685	1	685
110	681	1	681
111	675	1	675
112	669	2	1338
113	668	1	668
114	665	1	665
115	664	1	664
116	661	1	661
117	657	1	657
118	655	1	655
119	651	2	1302
120	649	1	649
121	646	1	646
122	645	1	645
123	644	1	644
124	643	2	1286
125	642	1	642
126	627	1	627

Rank	Freq.	Types	Tokens
127	609	1	609
128	604	1	604
129	601	1	601
130	597	1	597
131	595	1	595
132	594	1	594
133	588	1	588
134	580	2	1160
135	573	1	573
136	571	1	571
137	569	2	1138
138	568	1	568
139	567	1	567
140	560	1	560
141	558	1	558
142	555	1	555
143	554	1	554
144	552	1	552
145	550	1	550
146	544	2	1088
147	541	1	541
148	540	1	540
149	538	1	538
150	534	1	534
151	531	1	531
152	530	1	530
153	525	1	525
154	523	1	523
155	516	2	1032
156	508	1	508
157	502	1	502
158	500	2	1000
159	493	1	493
160	489	1	489
161	486	1	486
162	481	1	481
163	479	1	479
164	477	1	477
165	476	1	476
166	470	1	470
167	468	1	468
168	466	2	932
169	461	1	461
170	458	2	916
171	457	1	457
172	454	1	454
173	452	1	452
174	451	2	902
175	445	1	445
176	443	1	443
177	442	1	442
178	441	1	441
179	435	1	435
180	433	1	433
181	432	1	432
182	431	1	431
183	430	1	430
184	428	1	428
185	426	1	426
186	425	1	425
187	423	1	423
188	420	1	420
189	418	2	836
190	416	1	416
191	412	2	824
192	408	1	408
193	404	1	404
194	401	1	401
195	400	1	400
196	397	1	397
197	396	1	396
198	395	1	395
199	394	1	394
200	393	1	393
201	389	2	778
202	388	2	776
203	386	1	386
204	385	1	385
205	384	1	384
206	382	1	382
207	381	1	381
208	379	2	758
209	378	2	756
210	377	1	377
211	376	1	376
212	373	1	373
213	372	1	372
214	369	2	738
215	366	2	732
216	365	2	730
217	364	1	364
218	362	1	362
219	360	1	360
220	359	2	718
221	357	1	357
222	355	2	710
223	353	2	706
224	352	1	352
225	350	1	350
226	349	1	349
227	348	1	348
228	345	2	690
229	343	2	686
230	337	4	1348
231	334	2	668
232	332	1	332
233	331	2	662
234	328	2	656
235	327	1	327
236	325	1	325
237	324	2	648
238	323	1	323
239	322	2	644
240	321	1	321
241	320	1	320
242	319	1	319
243	318	1	318
244	316	1	316
245	315	3	945
246	314	1	314
247	313	1	313
248	312	1	312
249	311	1	311
250	310	3	930
251	306	1	306
252	304	1	304
253	303	3	909
254	299	1	299
255	298	2	596
256	297	1	297
257	296	3	888
258	294	2	588
259	293	1	293
260	292	2	584
261	291	3	873
262	290	3	870
263	289	2	578
264	288	2	576
265	287	1	287
266	286	1	286
267	285	1	285

Rank	Freq.	Types	Tokens
268	284	1	284
269	283	3	849
270	282	1	282
271	281	2	562
272	280	1	280
273	279	2	558
274	278	1	278
275	277	1	277
276	276	2	552
277	275	1	275
278	274	1	274
279	273	2	546
280	272	3	816
281	271	3	813
282	270	3	810
283	269	2	538
284	268	1	268
285	267	4	1068
286	265	2	530
287	264	1	264
288	263	3	789
289	261	2	522
290	260	2	520
291	259	2	518
292	258	6	1548
293	257	3	771
294	256	1	256
295	255	2	510
296	254	1	254
297	251	1	251
298	250	1	250
299	249	1	249
300	248	1	248
301	247	2	494
302	246	1	246
303	244	3	732
304	243	1	243
305	242	3	726
306	240	4	960
307	238	2	476
308	237	1	237
309	235	1	235
310	234	1	234
311	233	3	699
312	232	3	696
313	231	3	693
314	230	3	690

Rank	Freq.	Types	Tokens
315	227	1	227
316	226	6	1356
317	225	1	225
318	224	4	896
319	223	1	223
320	222	1	222
321	221	2	442
322	220	2	440
323	218	3	654
324	216	3	648
325	215	3	645
326	214	2	428
327	213	2	426
328	211	1	211
329	210	4	840
330	209	1	209
331	208	2	416
332	207	1	207
333	206	4	824
334	205	2	410
335	204	1	204
336	203	2	406
337	202	1	202
338	201	2	402
339	199	3	597
340	198	4	792
341	197	3	591
342	196	2	392
343	195	2	390
344	194	2	388
345	193	3	579
346	192	4	768
347	191	2	382
348	190	2	380
349	188	4	752
350	187	3	561
351	186	4	744
352	185	3	555
353	184	3	552
354	183	2	366
355	182	1	182
356	181	1	181
357	180	6	1080
358	179	3	537
359	178	3	534
360	177	6	1062
361	176	2	352

Rank	Freq.	Types	Tokens
362	173	1	173
363	172	2	344
364	171	3	513
365	170	7	1190
366	169	4	676
367	168	1	168
368	167	3	501
369	165	3	495
370	164	3	492
371	163	4	652
372	162	7	1134
373	161	4	644
374	160	9	1440
375	159	3	477
376	158	3	474
377	157	5	785
378	155	1	155
379	154	4	616
380	153	2	306
381	152	4	608
382	151	4	604
383	150	1	150
384	149	2	298
385	148	2	296
386	147	2	294
387	146	3	438
388	145	3	435
389	144	5	720
390	143	6	858
391	142	3	426
392	141	5	705
393	140	2	280
394	139	3	417
395	138	3	414
396	137	4	548
397	136	6	816
398	135	7	945
399	134	3	402
400	133	4	532
401	132	8	1056
402	131	4	524
403	130	3	390
404	129	3	387
405	128	12	1536
406	127	3	381
407	126	5	630
408	125	4	500

Rank	Freq.	Types	Tokens
409	124	2	248
410	123	8	984
411	122	6	732
412	121	2	242
413	120	5	600
414	119	3	357
415	118	4	472
416	117	5	585
417	116	5	580
418	115	7	805
419	114	11	1254
420	113	6	678
421	112	5	560
422	111	7	777
423	110	6	660
424	109	7	763
425	108	13	1404
426	107	4	428
427	106	7	742
428	105	7	735
429	104	11	1144
430	103	8	824
431	102	6	612
432	101	5	505
433	100	7	700
434	99	11	1089
435	98	9	882
436	97	5	485
437	96	2	192
438	95	8	760
439	94	17	1598
440	93	8	744
441	92	9	828
442	91	19	1729
443	90	11	990
444	89	11	979
445	88	14	1232
446	87	13	1131
447	86	15	1290
448	85	6	510
449	84	15	1260
450	83	14	1162
451	82	9	738
452	81	11	891
453	80	11	880
454	79	23	1817
455	78	17	1326

Rank	Freq.	Types	Tokens
456	77	16	1232
457	76	21	1596
458	75	13	975
459	74	20	1480
460	73	14	1022
461	72	20	1440
462	71	23	1633
463	70	24	1680
464	69	15	1035
465	68	22	1496
466	67	18	1206
467	66	18	1188
468	65	16	1040
469	64	18	1152
470	63	19	1197
471	62	19	1178
472	61	19	1159
473	60	24	1440
474	59	21	1239
475	58	28	1624
476	57	39	2223
477	56	39	2184
478	55	27	1485
479	54	36	1944
480	53	31	1643
481	52	26	1352
482	51	39	1989
483	50	29	1450
484	49	30	1470
485	48	41	1968
486	47	36	1692
487	46	46	2116
488	45	38	1710
489	44	39	1716
490	43	37	1591
491	42	65	2730
492	41	34	1394
493	40	71	2840
494	39	60	2340
495	38	60	2280
496	37	47	1739
497	36	97	3492
498	35	63	2205
499	34	88	2992
500	33	89	2937
501	32	82	2624
502	31	73	2263

Rank	Freq.	Types	Tokens
503	30	117	3510
504	29	86	2494
505	28	103	2884
506	27	104	2808
507	26	146	3796
508	25	111	2775
509	24	176	4224
510	23	151	3473
511	22	212	4664
512	21	156	3276
513	20	231	4620
514	19	174	3306
515	18	314	5652
516	17	199	3383
517	16	415	6640
518	15	271	4065
519	14	502	7028
520	13	304	3952
521	12	761	9132
522	11	363	3993
523	10	908	9080
524	9	785	7065
525	8	1620	12960
526	7	935	6545
527	6	2967	17802
528	5	1685	8425
529	4	6526	26104
530	3	5157	15471
531	2	19204	38408
532	1	24590	24590
		72157	**767801**