STEPHEN D. DUMONT

TEXTUAL EDITION AND ANALYSIS USING THE MICROCOMPUTER

Ten years ago computers had practically no place in humanist research. Today it is hard to imagine an academic work written without the aid of a computer. It remains, however, that most scholars in the humanities use their personal computers primarily, if not exclusively, as word processors. This is the case even though the enormous power and storage capacities of present-day personal computers far exceed the demands of any standard word processor. Part of the difficulty, of course, is that the development of commercial software has been directed to business or scientific applications rather than to the needs of academics in the humanities, which focus, ultimately, on texts rather than on data. However useful dBase IV, WordPerfect, or PageMaker might be to humanists in some respects, they are useless for textual analysis or the automated production of a critical edition. While some specialized commercial packages do exist for textual and editorial applications, they are not widely used by academics. Perhaps the most popular commercial program for text retrieval and concording is WordCruncher[1]. Nota Bene, the most academically oriented word processor, also contains a limited text retrieval program called Text-Base[2]. For the

[1] WordCruncher is a text retrieval program having much in common with TACT, although it is in some respects less flexible. WordCruncher cannot display a distribution or collocation of search results and is limited to a three level, numerical mark-up. For a review of WordCruncher, see Rubin Rabinovitz, « WordCruncher, » *Byte* 12.13 (Nov. 1987) : 216-20. The company that has distributed WordCruncher, Electronic Text Corporation, is currently undergoing reorganization. The program and information is still available, however, from Daniel Williams, Instructional Applications, 193 TMCB, Brigham Young University, Provo, Utah, 84602, USA.

[2] The main drawback to Nota Bene's Text-Base program is that it does not display the location (e.g., page or line) of the retrieved texts. Text-Base does allow, however, full Boolean searches with « wildcard » characters. The Text-Base program

generation of lemmatized concordances there is Micro-OCP (Oxford Concording Programme)[3]. Lbase is a multilingual, linguistic analysis software for use with grammatically analyzed texts. It can be used with such standard collections as the TLG (*Thesaurus Linguae Graecae*) text archive on CD-ROM[4]. One of the largest and most versatile programs for the production of scholarly editions is TUSTEP (Tübingen System of Text-Processing Programs)[5]. Even these exceptions show that there is not a great range of commercial software available for academic textual analysis and editing. Where such software is available, it is likely to be more than the individual scholar, working on an IBM personal computer, needs or wants to learn for the task.

The intent of this paper is to present some easy to use, non-commercial software designed by academics specifically for textual editing and analysis, together with some simple techniques for using this software. This software was written at the University of Toronto, supported in part by a unique, five-year, co-operative agreement between the University and IBM Canada, Ltd. The agreement created the Centre for Computing in the Humanities (CCH) at the University for the purpose of bringing computer technology to teaching and research in the humanities. The aims of the agreement, the projects it supported, and the products it delivered are documented in Ian Lancashire, *Humanities Computing : The CCH Toronto-IBM Co-operative* (Toronto, 1990). This publication and the software here described are available virtually free of charge from their authors at the University of Toronto.

It should be stressed that what follows does not require any expertise in computers other than a basic knowledge of DOS. The software described below is easy to use and the techniques are simple or even obvious. Nevertheless, the results produced by this software are so-

is currently undergoing a major revision, including support for Greek, for the release of Nota Bene version 4.0 in the Fall of 1991.

[3] Micro-OCP is distributed by Oxford Electronic Publishing, Oxford University Press, Walton Street, Oxford OX2 6DP, UK.

[4] Lbase was developed by, and is available from, John Baima, Silver Mountain Software, 7246 Cloverglen Drive, Dallas, Texas 75249, USA.

[5] Developed by Prof. Dr. Wilhelm Ott, Universität Tübingen, Zentrum für Datenverarbeitung, Brunenstrasse 27, D–7400 Tübingen, Germany. TUSTEP is not a single program but a system of programs capable of concording, editing, collating, and typesetting academic texts. See Dr. Ott's paper in this volume.

phisticated and, to my knowledge, difficult or even impossible to obtain otherwise on a microcomputer. The lesson to be imparted is that a humanist, without any special knowledge of computers, can convert by means of this or similar software a personal computer from an expensive typewriter into an inexpensive, powerful tool for textual research and manipulation.

The main technique to be used with our software is what might be called an « integrated text environment ». The idea is to create a working environment of a few simple programs in which the selfsame text can be collated, edited, and analyzed so that the results can be directly incorporated into an article or book currently being written in a word processor. Such an environment is possible because most word processors and the programs about to be discussed both import and export ASCII (American Standard Code for Information Interchange) text files. Thus, the editing and textual analysis programs produce their results as files that can be called into a document being produced on a word processor. Conversely, the word processor can be used to produce texts that serve as input files for the editing and textual analysis programs.

There are three programs that will make up our « environment »: NORM, MTAS, and TACT. The word processor is a matter of choice, provided that it is able to save and read material as an ASCII text file. The relationship between our programs and a chosen word processor in an « integrated environment » is illustrated in Figure 1, « Integrated Text Environment »[6]. In brief, NORM prepares an edited text, while MTAS and TACT analyze and search it. Reading Figure 1 clockwise from the top, the diagram shows that a text entered in a word processor can be successively edited and analyzed, and the results of the editing and analysis can be called back into the word processor for incorporation into a document. The programs are linked or « integrated », because they all read and write ASCII files.

In the present case, there are two main advantages of such an integrated approach. The first is to relegate to the computer the tedious and error-prone tasks of lineation and lemmatization required in the production of an edited text. The second is to submit the information in the edited text and critical apparatus to searching, indexing, and statistical analysis. The overall effect is to bring together related tasks that are

[6] All figures referred to are contained in the Appendix to this article.

usually carried out in isolation. For example, in the course of writing the introduction to a critical edition, the author can immediately determine through TACT how many homoeoteleuta occurred and in what manuscripts. The list of these homoeoteleuta can then be directly imported into the introduction without retyping a word of the critical apparatus. Similarly, a bar graph display of how these homoeoteleuta are distributed in the apparatus can be generated and included in an introduction even as it is being written.

Let us now examine the actual steps involved in editing and analyzed a text in our « environment » by representing in a sequential way our same clockwise path. The sequence of steps is given Figure 2, « Steps in the Editing and Analysis of a Text Using NORM, MTAS and TACT ». Boxes drawn in heavy or double lines surround programs. Boxes with single lines enclose what the program produces or does. Reading down the diagram, we see that NORM is a text-editing utility that automatically produces a lineated text, lemmatized variants keyed by line number, and a second apparatus keyed by line number, used in this case for sources. MTAS and TACT are searching, indexing, and statistical analysis programs that can be used on the text and apparatus produced by NORM. The fourth program, MAKBAS, is what prepares a text to be read by TACT. What follows is a description of each of these programs with examples of their applications.

NORM

NORM is a simple text-editing utility that automatically produces a lineated text and apparatus[7]. NORM was inspired by a mainframe program called Text-Handler, written by Ms. Rosalie Dieteman of the computer science department at St. Bonaventure University, New York, for use by the Franciscan Institute in their preparation of the critical editions of the philosophical works of Duns Scotus. Text-Handler automatically produces a lineated text and a two-level apparatus (variants

[7] A description of NORM is found in Lancashire, *Humanities Computing*, p. 18-19. Copies of NORM, together with its documentation and sample text files, can be obtained by sending a 3.5 inch diskette (740K or 1.44 MB) to Stephen D. Dumont, Pontifical Institute of Mediaeval Studies, 59 Queen's Park Crescent East, Toronto, Ontario, M5S 2C4, Canada.

and sources) by line number that is placed at the foot of the page. Written in uncompiled Basic, Text-Handler proved to be too slow and awkward when translated for use on a personal computer[8].

Professor Norman Zacour, in co-operation with Pontifical Institute of Mediaeval Studies, agreed to write a text-editing program designed to achieve results similar to those of Text-Handler but with the limitations of the personal computer in mind. Professor Zacour took the rather unusual step of writing the code for NORM entirely in assembly language. The result was an extremely compact and fast program. NORM takes up only 17K of disk space and produces an edited text and apparatus at a rate of 10K per second on a 10MHz, AT-type personal computer. The program is named NORM for its author.

In brief, NORM reads a collated text and produces three files : a lineated text or the established text ; the variants listed by line number together with their lemmata ; a second apparatus, designed for sources, listed by line number. All line numbers and lemmata are automatically assigned by the program. These steps occupy items 1-6 on the chart in Figure 2. The first step (item 1) is the preparation of the collated text. This may be done in any word processor or screen editor as long as the collated text can be saved as an ASCII file. In a standard fashion, one enters the text of the manuscript used as the base of the edition and then collates the other manuscripts against it. In this collation the variants of the other manuscripts are simply entered in curly brackets next to their lemmata in the text according to a few simple rules[9]. NORM will assume that the lemma of a variant is the immediately preceding word, which is in fact most often the case. If, however, the lemma is two or more preceding words, one only need type a asterisk before the bracket and then enter the full lemma as it is to appear in the apparatus. The apparatus of sources is entered in a similar fashion, although the sources are surrounded by a « # » and a « $ » instead of curly brackets. An

[8] A revision of Text-Handler for the personal computer for use with Microsoft Word is currently under way by the Franciscan Institute. For information, contact Prof. Gerard Etzkorn, Franciscan Institute, St. Bonaventure University, St. Bonaventure, New York, 14778, USA.

[9] The rules, while simple and few, must be followed exactly or else NORM will not calculate the line numbers properly. The rules are given in the documentation to NORM contained in the file NORM.XPL (for Norm-Explanation).

example of such a collated text is given in Figure 3. In this example, only variant readings and not sources have been entered for the sake of simplicity. The variants have also been put in bold characters for visibility ; NORM does not do this for you. The sample of collated text is from Peter Thomae's *Quaestiones de ente*, which has actually been edited and published with the aid of NORM[10].

The second step (item 2 on Figure 2) is to save the collated text as an ASCII file. Most word processors have a simple command or function key for this. In WordPerfect, for example, it is Control-F5. By saving the collated text as an ASCII file, all printer commands and other encoded information specific to the word processor are removed. This step is necessary if NORM is to be universal, since it is impossible to anticipate all ways in which different word processors encode their documents. Of course, after NORM has run, the results text and variant files may be retrieved in a word processor and the appropriate printer commands (e.g., underlining, superscripting, etc.) entered.

At this point, NORM is ready to run (item 3 in Figure 2). To run NORM, simply enter NORM at the DOS prompt followed by the name of the collated text file. As indicated, NORM produces the various files from the collated text very quickly. NORM informs the user of its progress and when finished displays the screen printed in Figure 4. As this figure shows, NORM tells the user that it has produced three files, TEXT.NEW, VARIANTS.NEW and FONTES.NEW, and indicates the length of each file. In this example, the FONTES.NEW file is empty since, as mentioned, no sources were included in the original collation file. Notice also that a Norton stopwatch has been used to time NORM, and the result is registered at the lower right hand corner as 13 seconds for a document of 130K, including text and apparatus. The output files of NORM are represented by items 4, 5 and 6 in Figure 2.

Actual samples of the output files TEXT.NEW and VARIANTS. NEW are found in Figures 5 and 6. Notice, by comparing the NEW texts with the collated text on page 4, what NORM has done. It has removed all the variants between the brackets together with the preceding word as the lemma, assigned the proper line number to the variant,

[10] S.D. DUMONT, *The Univocity of the Concept of Being in the Fourteenth Century II : The De ente of Peter Thomae*, in *Mediaeval Studies*, 50 (1988), p. 186-256.

reproduced the lemma, and separated the lemma from the variant reading and critical remark with a square bracket. In cases where more than one variant occurred on the same line, the variants are separated by a semicolon. Finally, NORM has lineated the text and placed hard returns at the end of each line. The hard returns fix the lines so that the lineation will not be affected when the text is retrieved in a word processor for insertion of printer codes[11].

On the other hand, notice what NORM has not done. Where a lemma is longer than a line, such as is the case with a long omission, NORM supplies only the line number of either the first or last word of the lemma, depending on where the variant is placed in the collated text[12]. The remaining line number must be entered manually. Secondly, NORM is not smart enough to detect if a lemma occurs more than once in the same line. This too must be noted manually. Also, the underlining or italicizing of the manuscript sigla and editorial remarks in the variant apparatus and of titles of works in the text is done in the word processor after NORM has run. This task, however, can be mechanically accomplished by means of a search and replace routine if all such material is enclosed within unique characters.

There are obvious benefits of NORM. One, clearly, is that NORM automates the process of setting up an apparatus and lineating a text. This in effect makes the process of editing a text, as it were, more plastic. Since, at any point in the editing process, the edited text and apparatus are only seconds removed from the collated text, changes can be made at will in either the text or apparatus without in reality disturbing the finished edition. The edition can be moulded and shaped with impunity by changing readings, reducing or expanding the apparatus, adding sources, and so forth. Since the collation can be done in a fully featured word processor, such changes themselves can be automated by means of global search and replace or macros.

Another obvious benefit of NORM is that it automatically produces from the collated text three files that can serve as textbases. The text and

[11] This of course assumes that the margins in the word processor are not set to fewer columns than one has set in NORM. One can vary the right margin in NORM by using DEBUG. See the NORM documentation file NORM.XPL.

[12] Quite obviously, one would want to enter the variant after either the first or last word of a lemma in a consistent way.

two apparatus can be thus analyzed by TACT and MTAS. Especially useful for such analysis is the edited text produced by NORM since the assigned line numbers can be exploited to generate absolute word indexes with no previous preparation of the text. The apparatus can be analyzed to determine the number of times manuscripts agree or disagree on variant readings.

MTAS

The acronym M-T-A-S stands for Microcomputer Text Analysis System. MTAS was designed and written by Ian Lancashire of the Centre for Computing in the Humanities at the University of Toronto and by Lidio Presutti of the University of Toronto Computing Service[13]. MTAS, as its name indicates, analyzes any text by searching, listing or graphing occurrences of letters, words or strings of words. MTAS has a complete help file that can be called to screen or printed out from a word processor.

MTAS is started by typing MTAS at the DOS prompt. The user is then met with the initial screen or main menu shown Figure 7. The desired function is chosen either by pressing the appropriate function key (F1-F7) or placing the cursor over the item and pressing RETURN. Thus to view the help files, one would press F1 ; to read a file on screen, F2 would be pushed, and so forth. The analytic programs are F3 (Word Frequency and Statistics), F4 (Word Distribution Graph) and F5 (Word Density Graph). While our concerns are limited to Word Frequency and Statistics (F3), MTAS's Word Distribution (F4) and Word Density (F5) graphing programs should be noted. MTAS will graph the occurrences of any word or word strings as they are distributed throughout the text. The graph will indicate on the y-axis the location of the word either by line number or by internal division embedded in the text. While the distribution graph shows how a word or expression is spread out over the text, the density graph gives a

[13] A description of MTAS is found in Lancashire's *Humanities Computing*, p. 144-145. MTAS is distributed by The Centre for Computing in the Humanities, University of Toronto, 14th Floor, Robarts Library, 130 St. George Street, Toronto, Ontario, M5S 1A1, Canada. Future plans for the revision of TACT include incorporating functions of MTAS.

measure of how expressions are clustered or crowded together. Between the distribution and density graphs, one can determine whether and where certain words or strings of words are clustered in a text. This is useful for tracing themes or concepts in a text. Because TACT performs these same functions in a more complete way, we shall defer discussion of them.

Our concern, however, is with MTAS's word listing and type/token statistics programs. As mentioned, to use these programmes one would press F3 as shown in Figure 7. MTAS then presents the user with another menu, shown in Figure 8. Reading down the menu in Figure 8, then, the word listing programs available are frequency word lists in alphabetical order (F2), reverse alphabetical order (F3), descending frequency order (F4), and a complete word index by line number (F5). The statistics programs are type/token analysis for both words and letters (F6). Let us apply some of these programs to our TEXT.NEW (Figure 4) generated by NORM.

First, MTAS can produce a complete word index by line number of the text. By selecting F5 (Figure 8), MTAS prompts the user for information : name of the file to be indexed and name of the file to which the index is to be written. It took MTAS about two minutes to read and produce a complete index for our sample text of about 80K. The first few entries of the word index are found in Figure 9. Notice that the line numbers of the index match the line numbers of TEXT.NEW generated by NORM. So, for example, the index shows that *abstractionem* occurs on lines 16 and *abstracto* on line 13. Comparison with TEXT.NEW in Figure 4 shows that these words do in fact occur on those lines in the text. In other words, because NORM has fixed the line numbers, MTAS can produce an absolute index of the text by line number. This is an example of an « integrated environment ». It is possible to move directly and quickly from a collated text, to an edition, to a complete index, which, since it is stored as a file, can be incorporated into a word processor. At no point has any of the text been manually modified.

MTAS can produce a word list sorted by frequency by selecting F2 (Figure 8), and again MTAS prompts the user for appropriate information. A sample of the word list generated by MTAS for TEXT. NEW is found in Figure 10. This frequency list can have various applications. For example, it can be used with the collocation feature in TACT to discover thematically significant clusters of words. It can be particularly useful when applied to the critical and source apparatus of an edition.

For instance, one could immediately determine how many times an author cited Aristotle or how many homoeoteleuta occurred in an edition. We shall shortly return to this latter application.

Finally, let us examine MTAS's ability to produce type/token statistics, F6 in Figure 8. MTAS defines the token/type distinction as follows. A word-token is a string bounded by word-separators, without respect to spelling. A word-type is a unique word, a string with a distinctive spelling. Thus, the number of word-tokens in a text is its total word count. The number of word types, on the other hand, is the number of different words in a text, that is to say, the vocabulary of the text. The Type/Token Statistics program in MTAS will produce a variety of tables, graphs, and statistical determinations concerning the frequency of both words and letters. The ability of MTAS to make these statistical determinations for *letters* will, as we shall see presently, have useful applications for analyzing critical apparatus. The main word type/token table produced by MTAS for our sample TEXT.NEW is given in Figure 11. At the top of the table observe that 1041 words occurred once, 303 words occurred twice, 168 words occurred three times, and so forth. The table also shows that the words with frequencies of one, two or three occurrences constitute together only 19.58% of the total words in TEXT.NEW. This would tend to indicate that TEXT.NEW has a very narrow and technical vocabulary, which is in fact the case. This can be confirmed from tables on the bottom of page 14. We see that TEXT.NEW has nearly 11,000 words or tokens but less than 2,000 different words or word types[14]. Taking into account that Latin is an inflected language, the vocabulary of TEXT.NEW is small. If these tests were performed on Boethius's *Consolation of Philosophy*, they would have produced very different results.

Type/token statistics are most useful for analysis of literary texts and poetry where richness of vocabulary is an important concern. Its

[14] Actually, these type/token figures are skewed because the line numbers, inserted by NORM into the margins of the text, have been included in the vocabulary of TEXT.NEW. Each of the more than 250 occurrences of line numbers (i.e., 5, 10, 15, etc.) would in this case count as a unique word form or a type. Thus, for accurate type/token ratios, the number of both types and tokens would have to be reduced accordingly. Ordinarily, roman and arabic numerals are excluded from the vocabulary of a text.

application to scholastic and technical texts is usually less revealing. Nevertheless, certain of the type/token features in MTAS can be manipulated to analyze the apparatus of a critical edition. To illustrate the application of MTAS to a critical apparatus, we will use the first question of Duns Scotus's *Quaestiones in Metaphysicam*, currently being edited at the Franciscan Institute[15]. This apparatus will provide a more complicated test case than the question of Peter Thomae in TEXT.NEW, which is found in only four manuscripts. The Franciscan Institute, as mentioned, uses a program similar to NORM to produce the edition of Scotus's philosophical works. It was thus possible to isolate the critical apparatus to Scotus's question using NORM.

The technique of using MTAS to analyze a critical apparatus involves two tricks. First, as far as MTAS is concerned, strings of manuscript sigla in the apparatus constitute words that can be counted and indexed just like any other word. Thus, in the case of a negative apparatus, the number of occurrences of a given « word » of manuscript sigla is the number of times those manuscripts agree in a variant reading against all other manuscripts. Second, as mentioned, MTAS is capable of giving type/token statistics for *letters*, which means that it can give such statistics for individual manuscript sigla since they are nothing but uppercase letters. Thus, with a little manipulation or preparation of the apparatus, MTAS can be used to count and graph affiliations between manuscripts, at least on the basis of their of readings as recorded in the apparatus[16].

[15] I am grateful to the editors, especially Rev. Romuald Green and Prof. Gerard Etzkorn, for allowing me access to the computer version of their edition prior to publication.

[16] Obviously, for MTAS to analyze manuscript sigla in a meaningful way, the apparatus must be prepared so that MTAS reads *only* the manuscript sigla and not other critical indications of the editor in the apparatus. This can be done in at least two ways. If the sigla are the only uppercase letters in the apparatus, MTAS can be set to a case sensitive sort. A second and neater way would be to remove all sigla to a separate file which MTAS would then read. This can easily be accomplished by bracketing all sigla in unique characters during collation. A macro can then be written to put all bracketed material into a separate file. One would, in any event, want to put sigla between unique characters during collation to facilitate entering underlining or other printer codes in a word processor after NORM has run.

For example, MTAS can produce an alphabetical frequency list of all manuscript groupings, such as that given for Scotus's question in Figure 12. Such a list enables the editor to find how many times any given group of manuscripts agree on a variant reading[17]. Perhaps more revealing is a list of all manuscript sigla and groups of sigla sorted by frequency, such as MTAS has produced for the same text in Figure 13. Here, for instance, it can immediately be seen that, since manuscript A occurs by itself in the apparatus 425 times, it differs from all other manuscripts 425 times[18]. That is, manuscript A has 425 isolated or unique variants, manuscript B has 236, and so forth. This information is valuable, for in this case manuscript A carries nearly twice the number of isolated variants as any other manuscript, suggesting possibly that A either belongs to an entirely different tradition or is a very inaccurate witness. By contrast, manuscripts C and M appear much more reliable. There is other information contained in the frequency sort in Figure 13. For example, since « 2 » in the Scotus edition is used to represent second hand corrections, we can tell that B is heavily corrected (98 times), while A is not (16 times).

It might even be more useful, however, to know not only the number of times each manuscript occurs in the apparatus just by itself, but also in combination with all other manuscripts. MTAS can provide this information because, as noted, it can calculate type/token statistics for *letters* as well as words. Thus, one of the type/token statistics for letters that MTAS provides is a « Frequency of All Letters ». This statistic for our sample apparatus of Scotus's question is given in Figure 14. According to these statistics, manuscript A occurs both alone and in combination with other manuscripts 701 times to constitute 12.08% of all variants in the apparatus. By contrast, manuscript C occurs only 224 times by itself or in combination with other manuscripts to form only 3.86% of the variants. Similarly, second hand corrections form 8.84% of the variants. Without type/token statistics for *letters*, such informa-

[17] This frequency list only gives the number of times each unique form of manuscript grouping occurs. To find out how many times a group occurs by itself *and* in combination with another group, one could specify this in a MTAS or TACT search.

[18] Once again, this assumes a negative apparatus, which is the case for Scotus's *Quaestiones in Metaphysicam.*

tion would be nearly impossible to determine. Another of the type/token statistics in MTAS useful for analysis of an apparatus is the « Word Length Statistic », given for our sample apparatus in Figure 15. This graph in Figure 15 shows that 2065 of all the manuscript sigla that occurred in the apparatus were words of one letter, that is, 2065 or 58.9% of all variants were carried by only one manuscript. Similarly, 27.7% of all variants were carried by two manuscripts or one manuscript corrected by a second hand, and so forth.

The above examples show how useful MTAS can be in combination with NORM. They illustrate well the concept of an « integral environment » for a text. Without ever retyping a word of the original collated text, one can produce a lineated edition, lemmatized apparatus, an absolute word index of the text by line number, frequency lists of both the text and apparatus, and statistical graphs and tables, particularly of the apparatus. All of these results are saved as files that can be manipulated in one's word processor. Further analysis and display of the same text is provided by TACT.

TACT

TACT is an interactive, textual analysis program written for the microcomputer by John Bradley and Lidio Presutti of the University of Toronto Computer Services[19]. TACT is a search and display program, also capable of graphing distribution and providing a statistical measure of coincidence of terms called collocation. While some of its search features overlap with those in MTAS, TACT can display the retrieved text is a variety of ways, all of which can be saved as files. What is more, TACT is capable of searching and displaying the distribution of « meta-textual » categories.

Before TACT can search and display a text, the raw text must be converted into a data base that TACT can digest. This preliminary step

[19] A description of TACT and illustrations of its use for thematic analysis of literary texts is found Lancashire, *Humanities Computing*, p. 147-171. TACT and its documentation are distributed by The Centre for Computing in the Humanities, University of Toronto, 14th Floor, Robarts Library, 130 St. George Street, Toronto, Ontario, M5S 1A1, Canada.

is achieved by a program called MAKBAS, short for MAKE-A-TEXTBASE. See items 9 and 10 in Figure 2. Besides indexing the text so that TACT can search it quickly, MAKBAS permits the user to define the structure of the text that TACT will search. If MAKBAS is run on a text for use with TACT without defining any structural elements or divisions in the text, TACT will only be able to display the results of its searches by line number. Properly used, however, MAKBAS can inform TACT of further divisions of the text into books, chapters, distinctions, articles, solutions, objections, replies and so forth. MAKBAS assumes, of course, that these divisions have actually be inserted into the text or, as it is termed, that the text has been « marked-up »[20]. The advantages of a structurally marked-up or encoded text are obvious. For example, it would be possible to determine in a properly marked-up text of Aquinas's *Summa*, how often a given authority, such as Augustine, is cited in objections as opposed to replies. In general, a more detailed textual mark-up will make possible more informative searches of the text.

After the text has been prepared with MAKBAS, it can be explored with TACT[21]. The main menu to TACT is found in Figure 16. The various actions, « Help », « eXit », and so forth, are at the top of the window, and they are chosen by pressing the space bar. TACT's display is an IBM standard called SAA, such as that used in dBase, and it employs a series of pull-down menus. We are interested in only two actions, « Select », which permits selection of the words or phrases to be searched in a text, and « Current », which displays the results of the search according to the currently chosen method of display. The pull-down menus for « Select » and « Current » are shown in Figures 17 and 18 respectively.

The Select menu given in Figure 17 shows that there are two basic ways in TACT to select words for searching. The first is to chose from the word list of the text (F3) and the second is to construct a search rule

[20] TACT does have, however, some automatic counting features for paragraphs and stanzas, for example. The ability TACT to handle virtually any mark-up scheme is one of its main advantages over WordCruncher.

[21] An additional step between MAKBAS and TACT may be required if the text to be used with TACT is large. TACT now is capable of linking several texts through its program called MERGEBAS. For details, see the TACT documentation.

(F2). Once a list or rule has been selected for TACT to use in its search, the results of the search can be display in the « Current » menu, shown in Figure 18. Notice the variety of ways in which the searched text can be displayed and that the parameters of any display can be modified (F4). Reading down the Current menu in Figure 18, TACT's displays are as follows :

KWIC : Key Word in Context
Text : Word with all the surrounding text
Index : Word with one line of text
Distribution : Graph of occurrences throughout the text
Collocate : Association of the selected word with other words

In Figures 19 and 20 are the two main ways of selecting text to be searched in TACT. Figure 19 shows the alphabetical frequency list. Here words can simply be marked directly from the list for TACT to search. A more useful and flexible way to search text is to write a search rule, as shown in Figure 20. TACT's syntax for search rules is modelled upon the UNIX « grep » command, which stands for « Globally look for Regular Expression and Print ». In Figure 20 a rule has been constructed using the regular expression symbols « . » and « * » to search for every form of the word *realitas*. TACT then displays the word list with frequencies selected by the rule, which is shown in Figure 21. Notice that TACT here provides the user with the opportunity to remove unrelated word forms, in this case perhaps *realiter*, from the list before TACT displays the results of the search. At this point, the results of the search can be displayed according to the « Current » display action, as indicated in Figure 18. Let us look, then, at all occurrences in our text of the forms of *realitas* according to the Index, Key Word in Context, Distribution, and Collocate displays.

The Index Display of our search is given in Figure 22. Note that the selected words are listed alphabetically with their frequencies beside them in parenthesis. In the Index display, as the name indicates, the occurrences are given with one line of text and the line number of the occurrence to the left. The same list displayed with Key Word in Context in Figure 23. Instead of the single line of text given by the Index Display, here we are given the context of five lines of text. The line number of the occurrence in the text is given at the right in parenthesis. As was the case with the word index given by MTAS, line numbers in both of TACT's displays are the same as those that NORM assigned to our sample TEXT.NEW. Although we have only displayed the location of the occurrence by line number, the display could be modified (F4) to give the location according to any internal division or divisions marked in the text (e.g., book, distinction, paragraph, etc.). What is more, both of these displays may be saved as files to be used in a word processor. Thus, for example, any passage in which the term *realitas* occurred could be directly placed in a footnote to a study on the text.

Finally, the Distribution Display is given in Figure 24 and the Collocate Display in Figure 25. The Distribution Display maps the rise and fall in the text of the selected expressions by means of a bar graph. The actual number of times an expression occurs within a percentage of the text is given, which is then represented graphically. The Collocate Display lists the words that occur nearby (usually defined as five words on either side) a chosen expression in the order of statistical significance measured by the Z-score. Roughly speaking, the Z-score is determined by comparing the frequency of actual co-occurrence of terms with the frequency of their co-occurrence assuming a random distribution. In general, the higher the Z-score, the more statistically significant a co-occurrence. The Z-score enables one to determine what terms are « clustering » in a text, thereby indicating what terms might be conceptually related. For example, one could use the frequency list produced by MTAS to determine the most frequently occurring substantive terms in a text. By using TACT's Collocate Display on those frequently occurring terms, one could determine what other terms clustered with them, possibly revealing the key concepts at work in the text.

I shall now illustrate how TACT has helped to solve a real problem in my own research, which concerns the philosophy of Duns Scotus, particularly his metaphysics. A notion central to Scotus's metaphysics is what he terms an « intrinsic mode or grade » of being. As important as this notion is to his metaphysics, Scotus gives it rather sparse development in his writings. It was my hypothesis that Scotus took the notion of intrinsic mode or grade, necessary for his account of transcendental being, from the categorical concept of intension and remission of accidental forms, which is typically said to occur in « grades ». This hypothesis, however, was difficult to test in Scotus's own writings because, as indicated, he mentions intrinsic modes infrequently. An early fourteenth-century text on Scotus's doctrine of transcendental being, written by an unknown but very faithful *scotellus*, did provide an opportunity to explore my hypothesis about the origin intrinsic modes[22].

One piece of evidence relevant to my hypothesis about the origin of the concept of an intrinsic mode was lexical. Did our *scotellus*, who on other matters was a very faithful interpreter of Scotus, give any preference, one way or another, to calling the intrinsic differences of being « modes » or « grades » ? That is, did our *scotellus* associate the concept of « intrinsic » with that of « mode » or « grade » ? Scotus himself appeared to hold no such preference, although he did not use the terms often enough to determine this firmly. One way of ascertaining whether terms are associated in a text is to examine their co-occurrence. Terms found often together are probably related in meaning and importance. In turn, one way of determining co-occurrence of terms is to map their distribution. Thus, terms possessing the same distribution in a text are likely to be related thematically.

Using TACT it was possible to plot the distribution for all inflected forms of the terms *gradus*, *modus* and *intrinsecus* in the rather lengthy *Quaestiones de conceptu entis* of our *scotellus*. The results, which are shown in Figure 26, were extremely clear. As is evident, the distribution of all forms of *gradus* and all forms of *intrinsecus* are nearly a perfect match. The forms of *modus*, on the other hand, are obviously unrelated. To ensure against the slight chance that *intrinsecus* and *gradus* were identically distributed not close to one another in the text, the distribution of *gradus* « nearby » (i.e., within five words) *intrinsecus* was displayed. This distribution was again identical to those of *intrinsecus* and *gradus*. These same results were confirmed by using TACT's collocation display for *intrinsecus*. This display is found in an abbreviated form in Figure 27. Here also the findings were clear. At the top of the list are the terms *gradu*, *gradum*, *gradus* and *gradui*, all with statistically significant Z-scores. By contrast, there is only one co-occurrence of any form of *modus*, and it has a statistically insignificant Z-score. All of this strongly suggested that our *scotellus* always tied the concept of « intrinsic » with that of « grade » and never with that of « mode ». Of course, the point of this example is not doctrinal but

22 S.F. BROWN and S.D. DUMONT, *Univocity of the Concept of Being in the Fourteenth Century III : An Early Scotist*, in *Mediaeval Studies*, 51 (1989), p. 1-129.

methodological. The above results were not obtainable manually but only by means of such programs as MTAS and TACT. These or similar programs make a text accessible in a way not otherwise possible.

TACT contains features that have not been explored here. For example, TACT can search and give the distribution for meta-textual categories, which are constructs of conceptually related terms. Also, the most recent version of TACT has a utility called MERGEBAS that allows the creation of large textbases. Finally, and most importantly, the figures of printed screens in the Appendix do not begin to convey the interactive and flexible nature of TACT's searching and display capabilities. One can move effortlessly from one search and display to another, probing and prodding the structure of a text in an enlightening way. After working with TACT, a printed concordance will seem by contrast static and rigid. The flexibility of TACT, however, can only be experienced by using it.

*

*   *

There will always be a role for mainframe computing in the humanities and for their large and complex task of managing enormous banks of textual data. The role of personal computers in scholarship, however, is perhaps even larger in the everyday work of the individual researcher. The aim of this paper has been to show that the role of the mircocomputer can be expanded considerably through a simple combination of easy to use, public domain software presently available.

Pontifical Institute of Mediaeval Studies, Toronto

APPENDIX OF FIGURES

FIGURE 1
Integrated Text Environment

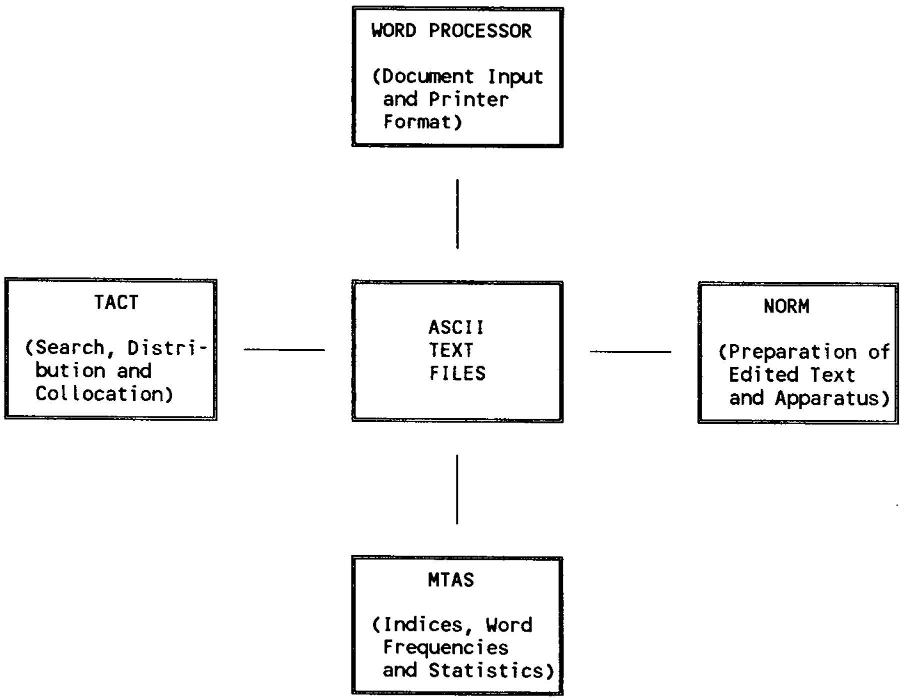

FIGURE 2
Steps in the Editing and Analysis of a Text Using NORM, MTAS, and TACT

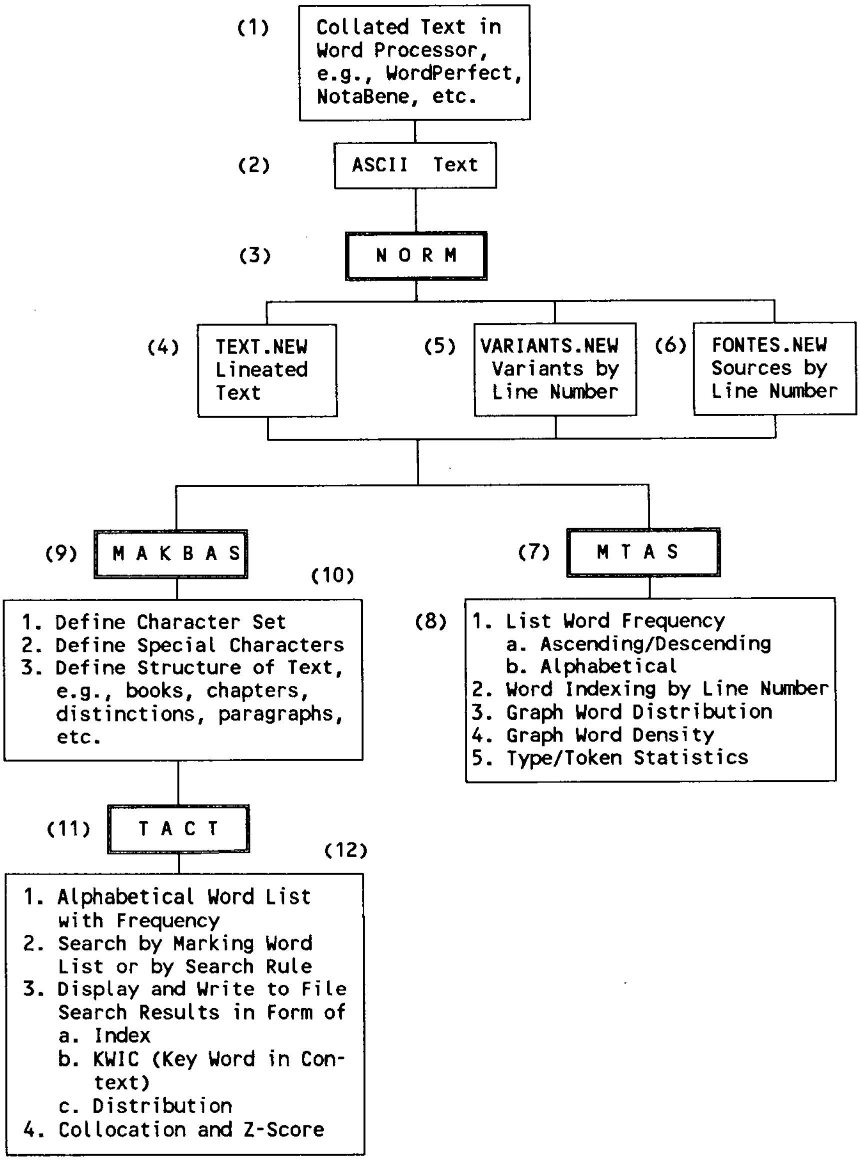

FIGURE 3
Sample Collated Text

Ad **{om. N}**tertium **{quartum O}**decimum sic proceditur: Et videtur quod ens praedicetur **{-catur O}**in quid **{om. V}**de ultimis differentiis, quia per hoc quod **{om. V}**philosophi erant **{fuerunt V}**certi de primo principio quod erat ens et dubitabant **{-tant O}**an esset hoc ens vel illud, concluditur **{-dit N}**secundum Scotum ens esse univocum primo principio et principiatis. Sed philosophi erant certi de ultima differentia quod***{quod...ens om. O}**erat ens et tamen non erant certi an esset **{erat O}**substantia vel **{aut W: om. N}**accidens. Ergo per eandem rationem mens est commune univocum dictum in quid de ultimis differentiis. Minor probatur aliqui enim istorum posuerunt **{ponunt O: om. V}**materiam esse totam substantiam **{rei add. O}**et secundum illos formae **{formas V}**substantiales a quibus sumuntur differentiae essent **{erant W: sunt O}**accidentia.

Praeterea omnis praedicatio in abstracto***{in abstracto] abstracta V}**est essentialis et quiditativa. Sed ista est vera `rationalitas est entitas'. Ergo entitas praedicatur quiditative de rationalitate. Maior **{Minor VW}**patet quia est secundum ultimatam **{-mam V}**abstractionem a conditionibus accidentalibus. Minor probatur quia rationalitas non est nihileitas. **{non entitas O}**Confirmatur quidquid essentialiter praedicatur de praedicato **{-mento N}**essentialiter praedicatur de subiecto. Sed entitas essentialiter et non per accidens est ens. **{entitas NV}**Ergo rationalitas essentialiter **{om. N}**est ens.

FIGURE 4
Appearance of Screen after Running NORM

```
C:\NORM> norm peterth.txt

                 NORM (Ver.3.0)

         TEXT.NEW                 VARIANTS.NEW
         85889 bytes              33624  bytes

                 FONTES.NEW
                  0  bytes

Copyright Norman Zacour 1990
C:\NORM> tm stop
                                9:54 am, Tuesday, August 14, 1990
                                                       13 seconds
C:\NORM>
```

FIGURE 5
TEXT.NEW Produced by NORM

Ad tertium decimum sic proceditur: Et videtur quod ens praedicetur in quid de ultimis differentiis, quia per hoc quod philosophi erant certi de primo principio quod erat ens et dubitabant an esset hoc ens vel illud, concluditur secundum Scotum ens esse univocum primo principio et principiatis. Sed philosophi erant certi de ultima differentia quod erat ens et tamen non erant certi an esset substantia vel accidens. Ergo per eandem rationem mens est commune univocum dictum in quid de ultimis differentiis. Minor probatur aliqui enim istorum posuerunt materiam esse totam substantiam et secundum illos formae substantiales a quibus sumuntur differentiae essent accidentia.

Praeterea omnis praedicatio in abstracto est essentialis et quiditativa. Sed ista est vera `rationalitas est entitas'. Ergo entitas praedicatur quiditative de rationalitate. Maior patet quia est secundum ultimatam abstractionem a conditionibus accidentalibus. Minor probatur quia rationalitas non est nihileitas. Confirmatur quidquid essentialiter praedicatur de praedicato essentialiter praedicatur de subiecto. Sed entitas essentialiter et non per accidens est ens. Ergo rationalitas essentialiter est ens.

Praeterea denominabile non praedicatur secundum se de denominativo nec determinabile de determinativo. Sed haec est secundum se `rationalitas est ens'. Ergo sic dicendo non est praedicatio denominativa. Ergo est praedicatio superioris de inferiori et per consequens univoci de univoco. Maior probatur

FIGURE 6
VARIANTS.NEW File Produced by NORM

1 Ad] om. N; tertium] quartum O
2 praedicetur] -catur O; quid] om. V; quod] om. V
3 erant] fuerunt V
4 dubitabant] -tant O; concluditur] -dit N
6 quod...ens om. O
7 esset] erat O; vel] aut W: om. N
10 posuerunt] ponunt O: om. V; substantiam] rei add. O
11 formae] formas V; essent] erant W: sunt O
13 in abstracto] abstracta V
15 Maior] Minor VW
16 ultimatam] -mam V
18 nihileitas] non entitas O
19 praedicato] -mento N
20 ens] entitas NV
21 essentialiter] om. N
23 denominativo] -tivis O: -nato V; nec] non N; de determinativo] determinato V
26 univoci] -cae W

FIGURE 7
Opening Screen or Main Menu for MTAS

```
                    Micro Text-Analysis ystem
                            for the
                  IBM Personal Computer System
                   (Version 2.0, March 1988)

Today's date: Friday April 26, 1991          Current time: 1:40:02 p.m.

Use UP and DOWN arrow keys to move, ENTER to select "Menu Option", ESC to exit.
                 ┌──────────────────────────────────┐
                 │  Micro Text-Analysis System      │
                 │                                  │
                 │ F1 Brief notes on MTAS programs  │
                 │ F2 Browse file on the screen     │
                 │ F3 List Word Frequency/Statistics│
                 │ F4 Graph Word Distribution       │
                 │ F5 Graph Word Density            │
                 │ F6 Misc. File Management         │
                 │ F7 Quit (EXIT from MTAS)         │
                 └──────────────────────────────────┘
```

FIGURE 8
List Word Frequency Statistics Menu for MTAS

```
                   Micro Text-Analysis System
                            for the
                  IBM Personal Computer System

                 Word Frequencies and Statistics
              ┌──────────────────────────────────────────────┐
Today's date: │F1 Brief notes on Frequency programs          │  :  1:40:02 p.m.
              │F2 Alphabetical Order                         │
Use UP and DO │F3 Reverse Alphabetical Order (by word endings)│ , ESC to exit.
              │F4 Frequency Order (High to Low)              │
              │F5 Word Index (with line numbers)             │
              │F6 Type/Token Statistics                      │
              │F7 Quit (EXIT from Freq/Stat menu)            │
              └──────────────────────────────────────────────┘
                  │ F2 Browse file on the screen      │
                  │ F3 List Word Frequency/Statistics │
                  │ F4 Graph Word Distribution        │
                  │ F5 Graph Word Density             │
                  │ F6 Misc. File Management          │
                  │ F7 Quit (EXIT from MTAS)          │
                  └───────────────────────────────────┘
```

FIGURE 9
MTAS: Word Index by Line Number

a
11, 16, 44, 58, 61, 67, 67, 120, 121, 123, 126, 127, 129, 129, 147, 151
208, 321, 323, 356, 359, 371, 404, 442, 443, 445, 446, 507, 509, 556
566, 567, 573, 574, 643, 651, 659, 663, 708, 723, 725, 745, 772, 773
870, 946, 1060, 1078, 1221
a'
149, 150, 305, 307
ab
43, 47, 64, 65, 121, 122, 122, 131, 131, 140, 274, 276, 320, 323, 325
326, 360, 360, 361, 547, 570, 573, 576, 577, 582, 631, 660, 706, 710
715, 866, 923, 925, 927, 932, 932, 945, 951, 951, 968, 988, 1057, 1109
1130, 1220, 1281
absolutae
729
absolute
99, 100, 445, 861, 878, 1299
absonum
603
abstractae
1067
abstractio
455
abstractionem
16, 546
abstractis
1079
abstractive
283, 518, 1062
abstracto
13, 142, 151, 226, 228, 235, 1033, 1050

FIGURE 10
MTAS: Word List in Frequency Order High to Low

418	est	95	quid	61	aliquid
361	et	89	conceptus	59	praedicatur
307	non	80	differentiae	58	modo
288	quod	79	entis	53	primo
255	in	79	quiditative	53	si
182	per	72	quia	53	vel
181	ens	68	ex	51	cum
173	de	68	secundum	49	a
146	ad	64	sunt	49	simpliciter
133	sed	63	nam	48	aut
130	differentia	63	potest	47	formaliter
123	ergo	62	ut		
102	se				

FIGURE 11
MTAS: Type/Token Statistics

Frequency Rank	Observed Frequency of Rank	Words in Frequency	Types Total	Tokens Total	% of Types	% of Tokens	% of word in Freq.
1	1041	1041	1041	1041	52.47	9.48	9.48
2	303	606	1344	1647	67.74	14.99	5.52
3	168	504	1512	2151	76.21	19.58	4.59
4	87	348	1599	2499	80.59	22.75	3.17
5	54	270	1653	2769	83.32	25.21	2.46
6	38	228	1691	2997	85.23	27.28	2.08
7	54	378	1745	3375	87.95	30.72	3.44
8	37	296	1782	3671	89.82	33.42	2.69
9	8	72	1790	3743	90.22	34.07	0.66
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
361	1	361	1983	10567	99.95	96.19	3.29
418	1	418	1984	10985	100.00	100.00	3.81

```
Number of Types   =      1984
Number of Tokens  =     10985
Type/Token ratio  =        0.181
Token/Type ratio  =        5.537
Hapax Legomena    =      1041
Hapax Dislegomena =       303
Hapax Legomena/Dislegomena ratio    =    3.4356
Hapax Legomena/Number of Types      =    0.5247
Hapax Legomena/Number of Tokens     =    0.0948
Hapax Legomena cubed/Types squared =  286.5952
Variance ( S.D. squared )           =  426.6956
Standard Deviation (S.D.)           =   20.6566
Coefficient of skewness             =   12.1160
Coefficient of kurtosis             =  184.4374
Herdan's characteristic             =    0.0838
Yule's characteristic               =  751.6013
Carroll TTR (Types / Sqrt of 2 X Tokens) =    13.3853
Most Frequent word "est" occurred 418 times
repeat rate (Tokens / frequency most frequent word) =    26.2799
```

FIGURE 12
MTAS: Alphabetical List of Manuscript Sigla for
Duns Scotus, <u>Quaestiones in Metaphysicam</u>, 1.1

425	A						
1	A'	1	ABDFJ	1	ABHJ	1	ACFHM
8	AB	1	ABDFM	1	ABHJL	1	ACGHJL
1	ABCDM	8	ABE	1	ABHM	1	ACGL
1	ABCEL	1	ABEFGHJ	3	ABJ	1	ACGM
1	ABCHL	1	ABEGHJ	3	AB2	1	ACH
1	ABCHLM	1	ABEGHJM	1	AB2DFGH	1	ACL
2	ABDEF	1	ABEHM	5	AC	1	ACLM
1	AB'DEFGHJM	1	ABELM	1	ACDEFGJM	4	AD
1	ABDEM	1	ABE2	1	ACDE2	1	ADE'F
1	ABDE2	3	ABGHJ	1	ACDF	1	ADEFGHJ
2	ABDF	1	ABGHJL	1	ACDFL	1	ADEFHM
1	ABDFGLM	2	ABGJ	1	ACEFLM	1	ADEFM
2	ABDFGM	2	ABH	1	ACFGJM	. . . .	

FIGURE 13
MTAS: Frequency List of Manuscript Sigla for
Duns Scotus, <u>Quaestiones in Metaphysicam</u>, 1.1

425	A						
236	F	46	D	11	ADFM	6	CL
227	H	34	GHJ	11	GH	6	DFHM
206	G	32	M2	11	J'	6	EF
167	J	30	HJ	10	AF	6	FG
154	L	29	G2	10	BEH	6	S
149	E	27	GJ	10	BH	5	AC
140	B	26	D2	10	EM	5	AM
112	C	22	AGHJ	9	AE	5	BEM
109	M	18	1	9	G'	5	BF
98	B2	17	DE2	8	AB	5	BG
91	BE	16	A2	8	ABE	5	BM
91	E2	16	E'	8	CH	5	CE
70	DFM	14	FM	8	DM	5	EH
69	DF	13	AG	8	F2	5	G1
56	J2	13	AH	8	3	5	H1
54	L2	13	H'	6	AGJ	5	LM
47	H2	13	I	6	CG	. . . .	

FIGURE 14
MTAS: All Letters Sorted by Frequency for
Duns Scotus, Quaestiones in Metaphysicam, 1.1

```
                                   Percentage
Letter Freq.    %          10       20       30       40       50
                 +----+----+----+----+----+----+----+----+----+----+
A       701  12.08 |************
F       579   9.98 |**********
H       570   9.82 |**********
E       550   9.48 |*********
2       513   8.84 |*********
B       508   8.75 |*********
G       507   8.74 |*********
J       483   8.32 |********
M       400   6.89 |*******
D       374   6.44 |******
L       339   5.84 |******
C       224   3.86 |****

Total all letters (Tokens)       =    5803
Total different letters (Types)  =     115
Type/Token ratio                 =       0.0198
Arithmetic Mean                  =      50.4609
Standard Deviation (S.D.)        =     152.1325
Herdan's characteristic          =       0.2811
Repeat rate for all letter "A" =         8.28
```

FIGURE 15
MTAS: Word Length Statistics for
Duns Scotus, Quaestiones in Metaphysicam, 1.1

```
Word   Freq.    %                    Percentage
Length                  10   20   30   40   50   60   70   80   90  100
                 +----+----+----+----+----+----+----+----+----+----+
   1   2065  58.90 |*****************************
   2    949  27.07 |**************
   3    278   7.93 |****
   4    116   3.31 |**
   5     61   1.74 |*
   6     26   0.74 |
   7      8   0.23 |
   8      2   0.06 |
   9      1   0.03 |

Total letters (Tokens)      =     5803
Total Words (Types)         =     3506
Type/Token ratio            =        0.6042
Mean word length            =        1.6552
Variance (S.D. squared)     =        1.0488
Standard Deviation (S.D.)=           1.0241
Herdan's characteristic     =        0.0104
```

FIGURE 16
TACT: Main Menu

```
Help (F1)  eXit (F10)  Select  Current  New  cAtegory  File          191 K

                TACT Version 1.2     (June 1990)

          Text:  (PETERTH.TDB)
          PDB : -----
          Number of Types: 1984         Tokens: 10985

            For help, push the F1 key. To exit, push F10.
        To choose action from Action Bar, push the SPACEBAR
Short Cuts:
F1: Help                      F5: Resize (^Move)     F9: Synchronize (^Zoom)
F2: Rule (^Import)            F6: Print (^File)      F10: QUIT
F3: Select (^WordList)        F7: Next Display (^Panel)
F4: Modify (^Close) Display   F8: Create Category
```

FIGURE 17
TACT: Select Menu

```
Help (F1)  eXit (F10)  Select  Current  New  cAtegory  File          191 K
                        selected List ... F3
                        Word List        ^F3
                        Categories
                        Rule ...          F2
                TA                          ne 1990)

          Text:  (PETERTH.TDB)
          PDB : -----
          Number of Types: 1984         Tokens: 10985

            For help, push the F1 key. To exit, push F10.
        To choose action from Action Bar, push the SPACEBAR
Short Cuts:
F1: Help                      F5: Resize (^Move)     F9: Synchronize (^Zoom)
F2: Rule (^Import)            F6: Print (^File)      F10: QUIT
F3: Select (^WordList)        F7: Next Display (^Panel)
F4: Modify (^Close) Display   F8: Create Category
```

FIGURE 18
TACT: Current Menu

```
Help (F1)  eXit (F10)  Select |Current| New  cAtegory  File           191 K
                             |Modify ...   F4|
                             |KWIC           |
                             |Text           |
                             |Index          |
              TACT Versi     |Distribution   | 1990)
                             |Collocate      |
                             |Next         F7|
         Text: (PETERTH.T    |Synchronize  F9|
         PDB : -----         |Close       ^F4|
         Number of Types:    |Panel       ^F7| ns: 10985
                             |Zoom        ^F9|
           For help, push                     it, push F10.
      To choose action from Action Bar, push the SPACEBAR

Short Cuts:
F1: Help                       F5: Resize (^Move)      F9: Synchronize (^Zoom)
F2: Rule (^Import)             F6: Print (^File)       F10: QUIT
F3: Select (^WordList)         F7: Next Display (^Panel)
F4: Modify (^Close) Display    F8: Create Category
```

FIGURE 19
TACT: Word List

```
Help (F1)  eXit (F10) |Select| Current  New  cAtegory  File            191 K

 a . . . . . . . . . . . . . .49
 a'. . . . . . . . . . . . . . 6
 ab. . . . . . . . . . . . . .46
 absolutae . . . . . . . . . . 1
 absolute. . . . . . . . . . . 6
 absonum . . . . . . . . . . . 1
 abstractae. . . . . . . . . . 1
 abstractio. . . . . . . . . . 1
 abstractionem . . . . . . . . 2
 abstractis. . . . . . . . . . 1

Short Cuts:
F1: Help                       F5: Resize (^Move)      F9: Synchronize (^Zoom)
F2: Rule (^Import)             F6: Print (^File)       F10: QUIT
F3: Select (^WordList)         F7: Next Display (^Panel)
F4: Modify (^Close) Display    F8: Create Category
```

FIGURE 20
TACT: Selecting by a Rule

```
Help (F1)  eXit (F10)  Select  Current  New  cAtegory  File            191 K

        AutoSelection Dialog Box
   Rule:
   realit.*

   Config Context:    Unit: word
                      Before: 5         After: 5

   For list of Reference Names for Unit, type "?" in the
                      Unit Field.

   Manual Select? Yes: X  No: _

Short Cuts:
F1: Help                       F5: Resize (^Move)      F9: Synchronize (^Zoom)
F2: Rule (^Import)             F6: Print (^File)       F10: QUIT
F3: Select (^WordList)         F7: Next Display (^Panel)
F4: Modify (^Close) Display    F8: Create Category
```

FIGURE 21
TACT: Results of the Search Rule realit.*

```
Help (F1)  eXit (F10)  Select  Current  New  cAtegory  File            191 K

        AutoSelection Dialog Box
   Rule:
   realit.*

> realitas. . . . . . . . . . .10
> realitate . . . . . . . . . . 8
> realitatem. . . . . . . . . . 4
> realitates. . . . . . . . . . 2
> realitati . . . . . . . . . . 3
> realitatis. . . . . . . . . . 2
> realitatum. . . . . . . . . . 1

F1: Help                       F5: Resize (^Move)      F9: Synchronize (^Zoom)
F2: Rule (^Import)             F6: Print (^File)       F10: QUIT
F3: Select (^WordList)         F7: Next Display (^Panel)
F4: Modify (^Close) Display    F8: Create Category
```

FIGURE 22
TACT: Index Display of Search for realit.*

```
realitas (10)
  (138)   est simpliciter simplex. Sed ultima >realitas seu perfectio |
  (147)    non est | ens formaliter, nam illa >realitas a qua sumitur non
  (149)           simplicem. Unde si | talis >realitas sit `a' haec non est
  (507)       cuiuslibet | differentiae quam >realitas a qua sumitur est
  (508)            simplex. | Contra. Omnis >realitas simpliciter simplex
  (508)     realitas simpliciter simplex est >realitas | divina. Sed
  (509)    simplex est realitas | divina. Sed >realitas a qua sumitur
  (510)      huiusmodi per | 510 te. Ergo est >realitas divina, quod est
  (581)       secundo quia dicit quod ultima >realitas est simpliciter |
  (654)          quae quidem non potest poni >realitas | 655 individualis,
realitate (8)
  (122)          quaedam | sumitur ab aliqua >realitate seu formalitate non
  (131)     re, sumitur tamen ab alia et alia >realitate seu | formalitate,
  (140)     et ideo | 140 differentia ab illa >realitate sumpta est ultima. |
```

FIGURE 23
TACT: KWIC (Key Word in Context) Display for realit.*

```
realitas (10)
    aliquid, hoc est, secundum aliquam realitatem, et secundum
    aliquam ignorari.Et per consequens talis naturae conceptus non
    est simpliciter simplex. Sed ultima realitas seu perfectio
    realis talis naturae omnino est simpliciter simplex, et ideo
    differentia ab illa realitate sumpta est ultima.
------------------------------------------------------------ (138)
    `rationalitas est ens', si rationalitas sit talis differentia.
          Quintum dictum quod differentia secundo modo sumpta non est
    ens formaliter, nam illa realitas a qua sumitur non includit ens
    quiditative, sed habet conceptum simpliciter simplicem. Unde si
    talis realitas sit `a' haec non est vera `a' inquantum `a' est
------------------------------------------------------------ (147)
    ens formaliter, nam illa realitas a qua sumitur non includit ens
    quiditative, sed habet conceptum simpliciter simplicem. Unde si
    talis realitas sit `a' haec non est vera `a' inquantum `a' est
    ens sed est per accidens et hoc sive `a' dicat illam realitatem
    sive differentiam in abstracto sumptam a tali realitate. Haec
------------------------------------------------------------ (149)
    inclusi non tamen possunt esse includentes, alias non essent
    praecise qualitativi. Teneo itaque quod tam conceptus cuiuslibet
    differentiae quam realitas a qua sumitur est simpliciter simplex.
          Contra. Omnis realitas simpliciter simplex est realitas
    divina. Sed realitas a qua sumitur differentia est huiusmodi per
------------------------------------------------------------ (507)
    praecise qualitativi. Teneo itaque quod tam conceptus cuiuslibet
    differentiae quam realitas a qua sumitur est simpliciter simplex.
          Contra. Omnis realitas simpliciter simplex est realitas
```

FIGURE 24
TACT: Distribution Display for realit.*

0-10%	3
10-20%	9
20-30%	0
30-40%	7
40-50%	9
50-60%	0
60-70%	0
70-80%	0
80-90%	0
90-100%	2

Total: 30

FIGURE 25
TACT: Collocate Display for realit.*

Collocates	Sel. Node	Collocate Freq	Type Freq	Z-score
divina	1	2	2	8.904
formalitate		2	2	8.904
realitatis		2	2	8.904
qua		4	8	8.687
competit		2	3	7.181
plures		2	3	7.181
sumatur		2	3	7.181
tali		2	3	7.181
alia		4	13	6.601
sumitur		4	14	6.320
aequivocatio		1	1	6.296
corporeitas		1	1	6.296
. . . . .	. . .	. . .	. . .	. . .
possint		1	2	4.342
quaeram		1	2	4.342
realis		1	2	4.342
ultima		5	45	3.776
sive		2	10	3.591

FIGURE 26
TACT: Distribution Displays of gradus, intrinsecus, and modus
Scotellus, Quaestiones de conceptu entis, Q.1

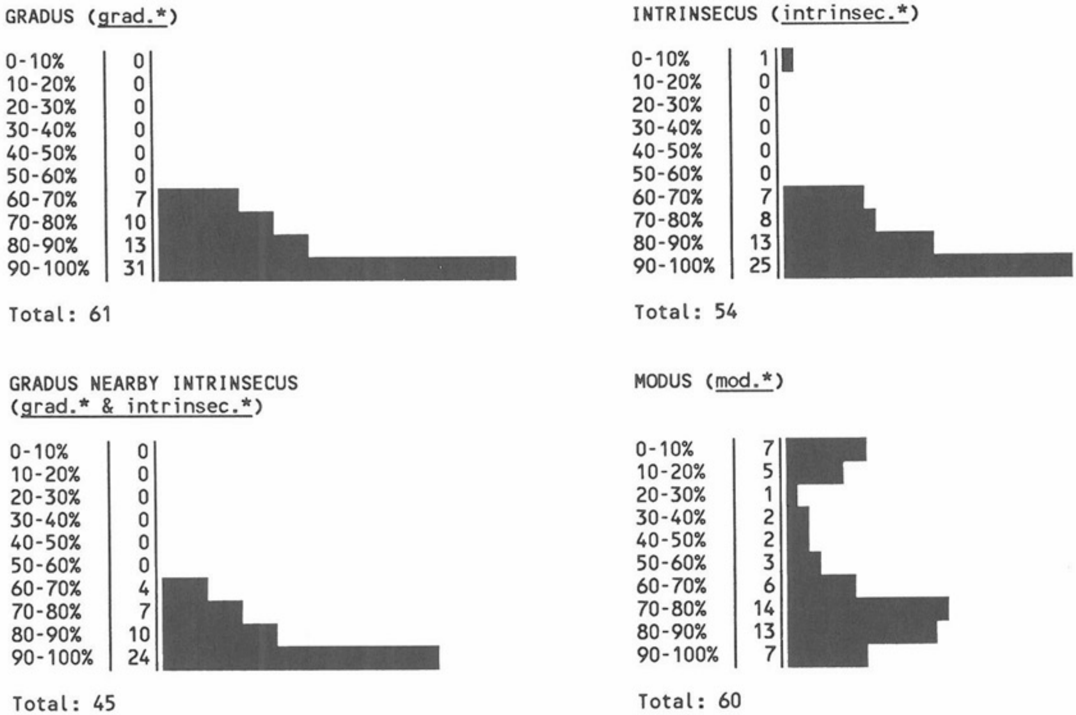

FIGURE 27
TACT: Collocate Display for intrinsecus
Scotellus, Quaestiones de conceptu entis, Q.1

Collocates	Sel. Node	Collocate Freq	Type Freq	Z-score
gradu		17	22	21.903
gradum		12	13	20.221
gradus		13	23	16.175
rei		12	28	13.329
naturae		10	21	12.905
sibi		15	53	11.722
realitatis		7	15	10.675
gradui		3	3	10.544
. . .	. . .	. . .	. . .	. . .
modum		1	17	0.856