Poster 1

STATE OF THE ART NUTRIENT DATABANK AVAILABLE TO PUBLIC VIA GERMAN VIDEOTEX SYSTEM; ALLOWS UPDATES OF COMMON PC SOFTWARE.

Roy Ackmann, Manfred Plath, Nutrition Information Center, University of Giessen, Germany.

Microcomputers and computer-aided nutrition programs have become vital tools for nutrition professionals and consumer educators, but they fall short in terms of timeliness and consistency. The Nutrition Information Center of the University of Giessen tackled the problem by using interactive Videotex (BTX) technology to create an easy-to-use online gateway from a user's PC to a state-of-the-art nutrient databank stored on a host computer. Users linked with the German government's Bundeslebensmittelschluessel (BLS) databank have access to analytical and computed ingredients of approximately 12,000 unique foods, including various recipe modifications and cooking preparations. The user can either use the BLS databank online to research food ingredients as in common food composition tables, or he/she can download the BLS data onto a microcomputer to update existing nutritional programs.

The BTX system is a new easy-to-use public communication services network accessed easily via normal telephone lines and displayed on any monitor. In addition to its gateway function, BTX offers brief and up-to-date nutritional information, access to literature references, and easy communication between the user and the Nutrition Information Center via electronic mail.

Poster 2

EXPANSION OF A NUTRIENT DATABASE FOR NATIVE ALASKAN FOODS

Sally Schakel, Barbara Pickering and I Marilyn Buzzard, Nutrition Coordinating Center, University of Minnesota and Elizabeth D Nobmann, Alaska Area Native Health Service, Anchorage, Alaska

A study of the relationship between the nutrient intake of the Siberian Yupik Eskimos and the incidence of diabetes, hyperinsulinemia and cardiovascular disease[1] required a nutrient database that included foods typical of this population. To meet this need, we expanded an existing nutrient database of American foods to include Native Alaskan foods. A pilot study of 92 24-hour recalls collected from Eskimos of St. Lawrence Island was used to determine foods typically consumed. From the recalls, a list of 40 foods not found in the existing database was compiled. Nutrient values for these foods were obtained mainly from the *Nutrient Value of Alaska Native Foods* by Elizabeth D Nobmann, Indian Health Service, Anchorage, and from scientific literature. Chemical analyses were performed on 14 foods for which nutrient values were not found in the existing literature.[2] Some foods were represented in the database by foods believed to be similar in nutrient content. For example, nutrient values for wild duck were used for loon, and values for duck eggs were used for murre eggs. Recipes for foods such as fried bread, agutuk and fishhead soup were provided by Alaskan nutritionists and added to the database. Before release of the database, quality control procedures were implemented to check for data accuracy and internal consistency within the database.

1 Ebbesson, S.O.E. Diabetes Risk Factors in Alaskan and Siberian Eskimos, NIH Grant DK-91-01 to the University of Alaska, Institute of Circumpolar Health

2 Funded by a grant from the Indian Health Service

Poster 3

COMPARISON OF USDA AND MANUFACTURER'S NUTRIENT VALUES FOR CONDENSED SOUP.

Carole R. Dichter, Bonnie Sherr, Chor San Khoo, Patricia Locket, Campbell Institute for Research and Technology, Camden, NJ

A comparison of USDA Handbook 8-6 (1979) nutrient values with a leading manufacturer's current mean values for the most popular varieties of condensed soup indicates significant differences and information gaps exist. For three soup varieties, Chicken Noodle, Cream of Mushroom and Tomato, the mean values for sodium on an equivalent weight basis (1 cup) are respectively 190 mg, 173 mg and 150 mg less per serving than USDA #8-6 published values. The mean values for fat and calories for Cream of Mushroom are also 25% and 19% lower respectively than the Handbook 8-6 values. New varieties of reduced sodium and low fat soups are not included in the 1979 data. The observed differences between published and mean values may reflect changes in product formulation and/or improvements in analytical methodologies. These findings underscore the need to use current brand name data for soup for national surveys, individual dietary assessments and meal planning.

Poster 4

DEVELOPMENT AND EVALUATION OF SOFTWARE FOR ESTIMATION OF NUTRIENT VALUES.

Brian J. Westrich, I. Marilyn Buzzard, Sally F. Schakel, and Paul G. McGovern, Nutrition Coordinating Center (NCC), University of Minnesota.

Software was developed that estimates nutrient values in commercial food products by mathematical optimization. The optimization methods used were linear programming (LP) and quadratic programming (QP). Dietary fiber and linoleic acid values were calculated from estimated ingredient amounts for 51 food products whose content of these two nutrients was known. Ingredient amount estimations were performed by three different nutritionists using three different methods – an existing trial-and-error method, as well as the LP and QP methods. Thus, a total of 459 ingredient amount estimations were made. Accuracy and efficiency of the three methods were compared via factorial analysis of variance. No statistically significant difference in accuracy was found between methods, but the time required to complete ingredient amount estimations using optimization methods was significantly less than the time required for the trial-and-error method ($p < 0.0001$). The degree of estimation bias, although similar for all three methods, varied as a function of the individual food products. Mathematical optimization techniques can increase the efficiency of maintaining nutrient databases.

Poster 5

MAXIMIZING DIETARY DATA QUALITY THROUGH APPLICATION OF QUALITY CONTROL FINDINGS: THE MDRD PHASE 3 STUDY EXPERIENCE.

Monica E. Yamamoto, Frani M. Averbach, Arlene W. Caggiula, Bonnie P. Gillis, Fran L. Jones, Rebecca Meehan, JoAnn Naujelis and the MDRD Study, MDRD-Nutrition Coordinating Center (NCC), University of Pittsburgh, Pittsburgh, PA.

Research studies conducted over several years typically use internal and external quality control (QC) procedures to monitor data quality. While this effort can prevent repetition of identified problems, archived data are potential reservoirs of these same problems. The Modification of Diet in Renal Disease (MDRD) Phase 3 Study, a large five-year clinical trial, implemented dietary data cleaning which incorporated QC findings. Our preliminary experience (1989-92 data) with this procedure is reported here.

The MDRD Study, sponsored by the National Institutes of Health and the Health Care Financing Administration, is designed to determine whether controlled dietary protein and phosphorus intake and/or blood pressure control will alter the progression of chronic renal disease. Search of archived MDRD dietary datasets (food records=10,861) located 1793 patient food records with food or portion choices identified through QC as potential problems. Original patient food records were retrieved from the files and codings reviewed. About 60% (n= 1075) of food records pulled required coding fixes. About 20% of these required multiple fixes. Our findings indicate that dietary data cleaning procedures which included quality control findings were important for MDRD data quality and are likely to be important for other research studies as well.

Poster 6

A COMPARISON STUDY BETWEEN TWO NUTRIENT DATABASES.

Cynthia S. Nicholson, Kathleen M. Koehler, Sharon J. Wayne, and Philip J. Garry, The University of New Mexico School of Medicine, Albuquerque, NM 87131

We compared the results of two nutrient databases as part of the adoption of a new nutrient database for the New Mexico Aging Process Study. Thirty-one 3-day food records, randomly selected from 250 records collected in 1991, were coded and analyzed using The University of Texas, Health Sciences Center's Food Intake Analysis System (FIAS), ver.1.0, 1990 and the Case Western Reserve University Nutrient Database System (CWRU), Rel.#10, 1989. Records selected were for 13 men and 18 women aged 66-88 years. All records were coded by the same trained nutritionist to control for intercoder variability. Daily totals and 3-day averages were compared for energy and 26 nutrients. Skewed distributions were transformed to achieve normality. The mean energy intake using FIAS and CWRU was 1686 kcal/d and 1656 kcal/d, respectively. The group means were not significantly different by paired t-tests for daily totals and 3-day averages for energy, carbohydrate, protein, cholesterol, fiber, total vitamin A, phosphorus and iron. The remaining 18 nutrients were significantly different ($p=<.05$) from each other in daily totals or 3-day averages or both. The FIAS output tended to be higher for all nutrients when compared to CWRU. Correlation coefficients for 27 nutrients ranged from 0.75 to 0.95 for the three day averages, except total tocopherol where $r = 0.57$. Possible explanations for the differences observed include missing values associated with the CWRU database, different version or release dates for the two databases, and differences in the number of food items available to make accurate coding decisions. (Supported by NIH AG-02049.)

Poster 7

DOCUMENTING ENTRY SUBSTITUTIONS WITH SIMILAR-FOOD CODES: EXPERIENCES FROM THE MODIFICATION OF DIET IN RENAL DISEASE (MDRD) STUDY.

Fran L. Jones, Frani M. Averbach, Arlene W. Caggiula, Bonnie P. Gillis, Rebecca J. Meehan, JoAnn A. Naujelis, Monica E. Yamamoto, and the MDRD Study, MDRD-Nutrition Coordinating Center (NCC), University of Pittsburgh, Pittsburgh, PA.

Foods consumed by Study patients are not always an exact match with database foods. The MDRD Study uses a coding system for substitutions, a "similar" field, which identifies all foods used for substitutions and specifies the original substituted food consumed. Tracking of coding substitutions is important for the several reasons which include: ensuring coding consistency; allowing for possible modification of Study datasets when precise information becomes available; and identifying foods needed for database updates. No reports are currently available on the magnitude of coding substitutions required for processing dietary data, so the MDRD Study's experience was examined for this purpose.

The MDRD Study is a nationwide multicenter clinical trial, sponsored by the NIH and HCFA, designed to determine whether the control of dietary protein and phosphorus intake and/or the reduction of blood pressure to two target levels will reduce the rate of progression of chronic renal disease. MDRD patients' diet prescriptions can be complex and their food choices often include modified usual and favorite foods, as well as newly available foods from the market place. During the MDRD Study baseline period, 2633 food records containing 139,429 food items were analyzed, 9.5% of those foods required coding substitutions. By mid-March 1993, 9,009 follow-up food records had been processed and 7.8% of the 584,784 foods reported required a similar code. Foods most often requiring coding substitutes were bakery items, fast foods, convenience items (soups and mixed dishes), and modified foods such as low fat cheeses. During follow-up, frequently reported foods requiring similar tags (e.g. 1½% low fat milk) were assigned coding guidelines to help ease data entry. Given the magnitude of coding substitutions required (nearly 60,000 food items), the ability to document their occurrence was critical to the quality of MDRD Study data.

Poster 8

DISCREPANCIES BETWEEN DIETITIAN AND CLIENT CALCULATIONS OF PROTEIN INTAKES.

Catherine A. Chenard and Linda G. Snetselaar, General Clinical Research Center and College of Medicine, University of Iowa, Iowa City, Iowa.

In a study of protein intake biomarkers, 12 normal volunteers consumed a self-selected 0.6 gram protein per kilogram standard body weight (SBW) diet for six days. Subjects kept daily food records and calculated their protein intake using food label information and patient materials designed for a multicentered trial, the Modification of Diet in Renal Disease (MDRD) study. Two dietitians independently calculated protein content of the 72 food records using the Minnesota Nutrition Data System. Protein intake calculated by both dietitians was higher (mean 11%; median 8.5%) than calculated by subjects. To determine reasons for this difference, the twenty-four food records with subject and dietitian discrepancies of 0.10 gram protein/kg SBW or greater were examined. Factors contributing to the discrepancies included protein values substituted for foods not found in food tables, insufficient food descriptor or portion size information recorded by subjects, differences among protein values found in food tables and on food labels, and food selection, portion size and protein calculation errors. Quality control of food record coding is important. Client training should include procedures for determining protein content of foods not listed in food tables, and calculating protein content of portions eaten.

Poster 9

USING FOOD RECORDS TO IDENTIFY EATING PATTERNS OF WOMEN AND MEN

BJ Scott and ST St.Jeor. Nutrition Education and Research Program, University of Nevada School of Medicine, Reno, NV 89557

The understanding of eating patterns and their relationship to overall dietary intake may provide important clues to diet-disease relationships. The purpose of this study was to examine one dimension of the eating patterns of both normal weight (N) and overweight (O) individuals. Subjects (S's) were women (W) (N n=25; O n=25) and men (M) (N n=26; O n=24) in their 20's who were participants in the RENO Diet Heart Study. Data sets were developed from detailed 7 day food records using a relational data base program and the USDA Standard Reference nutrient data set. Number of eating incidents (EI)(defined as food intake ≥ 30 minute intervals and/or at different locations) was used to compare S's by gender and weight classification and to examine potential relationship to diet and selected indices of CVD risk. The mean number of EI did not vary significantly by gender or weight, but men tended to eat slightly more frequently than women (5.7 vs 5.6), and obese S's ate more frequently than normal weight S's (OW=5.7; NW=5.4; OM=6.0; NM=5.5). The women and the NW tended to eat more frequently during the weekdays (Mon-Fri) than on weekends (EI: NW=5.6 vs 4.9; OW=5.8 vs 5.6: NM=5.6 vs 5.1)), while the OM had the opposite pattern (EI: 5.9 vs 6.1). The relationship of EI to calorie intake was examined, and significant positive correlations were found between EI and caloric intake for the O S's ($p \leq .01$). This was found for NM ($p \leq .05$) to be true only for the week ends. No relationship was found between EI and calorie intake for NW. Differing relationships were found between EI and % of calories from fat by gender and weight: inverse correlations were found for both NW ($p \leq .05$) and OW ($p \leq .01$), a positive relationship was noted for OM ($p \leq .05$) and no relationship was found for the NM. Overall, the OM tended to demonstrate the greatest correlations between diet and EI. Few significant ($p \leq .05$) correlations were found between EI and thirteen selected measures of CVD risk: NW – weight fluctuation (retrospective and prospective); OW – resting energy expenditure; and OM – systolic blood pressure. Further study and description of eating patterns and corresponding measures of meal composition and food choices may yield further insight about their relationship to risk factors for disease.

Poster 10

CONSISTENCY OF DIETARY DATA ENTRY USING MICROCOMPUTER SOFTWARE

Frani M. Averbach, Arlene W. Caggiula, Bonnie P. Gillis, Fran L. Jones, Deborah M. Larsen, Monica E. Yamamoto. University of Pittsburgh, Department of Epidemiology, Pittsburgh, PA.

Patient intakes are increasingly being evaluated with various types of software. To examine intake estimate consistency, one food record was entered twice by two research nutritionists with considerable coding experience using two versions of a single microcomputer program, N3 and N4. Consistency of entry included foods chosen, calculations made and resources used to support appropriate and accurate selections. Three important areas were identified that showed differences in entry. First, when several similar food items (same item description) are available on the database, how is one item selected? Second, are calculations performed by the practitioner and then entered or are exact amounts from food records entered leaving the calculations to the program? Third, when common measures or amounts recorded on food records don't match what is available, what resources are used to compute the quantities eaten? In conclusion, there are several areas where individual practitioners would have difficulties in making consistent decisions. Development of a users guide book for dietary data entry calculations and food selections, would decrease the inconsistency in entry, and increase patient compliance by providing accurate feedback.

Poster 11

TRANSFERRING WORD PROCESSOR FILES BY ELECTRONIC MAIL

John C. Klensin, Ph.D., INFOODS Secretariat

The use of programs such as WordPerfect (tm) to prepare documents and the use of electronic mail to support collaborative work are both increasing. As they do, there is increasing need to be able to combine the two, i.e., to send documents over wide area computer networks with the formatting preserved.

This informal poster will present the reasons why such document transfer, and transfer of other "non-ASCII" materials, has been a problem. Tools for encoding and decoding such documents and their strengths and weaknesses will be shown. Copies of public, or otherwise free and substantially unrestricted, versions of these tools will be available for people to copy onto their own diskettes.

Transferring Word Processor Files by Electronic Mail

John C. Klensin*

Abstract

The use of programs such as WordPerfect® to prepare documents and the use of electronic mail to support collaborative work are both increasing. As they do, there is increasing need to be able to combine the two, i.e., to send documents over wide area computer networks with the formatting preserved.

This poster presents the reasons why such document transfer, and transfer of other "non-ASCII" materials, has been a problem. Tools for encoding and decoding such documents and their strengths and weaknesses will be shown. Copies of public, or otherwise free and substantially unrestricted, versions of these tools are available from a number of locations.

The Problem

Ordinary electronic mail (the term "email" is used interchangeably below) is intended for transferring relatively unformatted messages in "ASCII text" ("EBCDIC text" on BITNET). These text forms are suitable for conventional, English-language, texts. They become problematic when characters must be included that don't appear in ordinary English, such as the ñ of Spanish, the ç of French, or any of the characters of Greek, Russian, Chinese, and so on. A number of conventions have sprung up for dealing with these characters over the years; none have been really satisfactory or generally accepted. Perhaps more important to those of us who write predominantly in English, ordinary electronic mail cannot preserve the subtle formatting information of microcomputer-based word processor files: notions of "hard" and "soft" line breaks, changes in fonts and sizes, differences between indented text and margin changes, all must be represented by conventions and then re-translated at the receiving end or lost entirely.

This problem occurs because few microcomputer-based word processing files consist of ordinary text. They contain extended "characters" to denote formatting and font changes and for other purposes. Several of them handle "soft carriage returns" by treating each paragraph as a single long line; many email systems cannot transfer such long lines.

Enhancements to email standards completed and approved within the last year and now gradually being propagated around the world are likely to provide general solutions to these problems in the long term (see "The Future", below). But, in the near term, if one wants to transfer word processor files, it is necessary to reach agreements between sender and recipient and then "code" and "decode" the files using special tools. The coding procedures make word processor files unintelligible without decoding, but forces all of the information into a form that can, with a bit of luck, be transmitted by electronic mail. The methods discussed below will continue to work even when future email enhancements make simpler alternatives available.

---

* INFOODS Secretariat, UNU, PO Box 500 Charles Street Station, Boston, MA 02114-0500. Tel: 617 227 8747. Fax: 617 227 9405. Email: Klensin@INFOODS.UNU.EDU

The author would like to acknowledge suggestions made by Sallie Fellows of the NH Department of Education for the presentation in this paper and the many people with whom these procedures have been worked out, especially Eric Poortvliet of the Wageningen Agricultural University in the Netherlands and Barbara Burlingame of the Crop Crown Research in New Zealand for their efforts and patience. Errors and confusion parts remain the responsibility of the author.

General Procedure

The procedures discussed here are presented in terms of WordPerfect for MSDOS®, since it seems to be the word processing tool most heavily used in the nutrient composition community. The general principles and coding and decoding tools are equally applicable to Microsoft Word® and other tools as long as the word processing file is self-contained (in WordPerfect, this means that you must be quite careful about, e.g., graphic boxes that reference external files). However, you must also be sure that your recipient can read whatever word processing format you use for your document.

In the Apple Macintosh® the procedures for transferring files will be similar, but the programs used will be Mac programs. The most widely-used Mac programs for encoding and decoding (steps S.3 and R.3 below) are StuffIt or other programs that produce the BinHex format.

Sending:

Before you begin, discuss with your partner the word processing package and versions you will use, the encoding methods you will use, and if you will use compression (see below). Since you must both have compatible programs (or very good conversion tools, planning ahead will save time and avoid frustration.

Step S.1: Prepare the WordPerfect file as you usually would. Note that your recipient will almost certainly have a different printer and set of fonts available (this is just how it works), so it is a good idea to give careful consideration to formatting as your document is composed. In particular, indent text with the [Indent] command, not with hard returns and tabs. Use hard page breaks only when they are strictly necessary; [BlockProtect] and [Conditional EOP] controls are strongly preferred. If you must use page numbers, indices, or tables of contents, set these up with WordPerfect facilities, not by typing page numbers in. When possible, change font size and attributes, rather than changing the base font. Your recipient will thank you. If he or she is going to edit the file and send it back, you will be much happier also.

Step S.3: Select a coding method (several are discussed below). You must choose one that is available to both you and your recipient. Use it to convert the file to coded form.

Step S.4: If necessary, transfer the now-coded file to the computer from which you send and receive electronic mail. Get it into format that you system uses for outgoing messages. Include in that message, or send separately, a note that indicates the word processing package and version, the encoding method, and, if the file has been compressed, the compression method. If you have worked things out in advance, this need not be complicated—a line like "WordPerfect 5.1, ZIPped, and BOOed" is typical—but it provides a use reminder for both of you. Then mail the message, possibly with a subject line that repeats these details.

Receiving:

Step R.1: Read the mail message and transfer it to a form from which you can download it to your PC. Then do either step R.2a or step R.2b.

Step R.2a: Download the file to your PC. Use a "program" or "ASCII" editor to delete the mail headers and other clutter. The best built-in editor of this type in MSDOS 5.0 or later is called EDIT and earlier versions provide EDLIN, but any "programmer's" or line editor will do.

If you prefer to use WordPerfect for this editing, follow these steps: (i) set very wide document margins. (ii) Read the file in as an ASCII file (TextIn/Out, DosText, 2). (iii) Delete the headers and other clutter, then save the "clean file" as an ASCII file (TextIn/Out, DosText, Save). Be sure that you don't overwrite the file you just created when you exit WordPerfect.)

--or--

Step R.2b: Edit the file on the machine where you received the email to get rid of the headers and other material. Then download the already-cleaned-up file.

Step R.3: Process the file with the appropriate decoder. If you can't figure out which decoder to use, check with the person who sent you the file. Trying to guess is feasible, but not often productive.

Step R.5: Read the decoded result into WordPerfect. Ignore messages about converting from a printer you have never heard of into your default printer; they are fairly normal (see the discussion under Step S.1 above).

The Compression Option

Alert readers will note that steps S.2 and R.4 seem to be missing above. They are optional steps, but, if the sender uses them, the receiver must too. They require additional matching software at both ends, and provide additional opportunities for things to go wrong. Hence they should be omitted: the procedures above will work without those steps.

WordPerfect files tend to be fairly large in ratio to the number of actual characters of text that will be displayed on the printed page. They become especially bloated if you use large style libraries, multiple printer setups, or complex formatting. Many electronic mail systems (especially BITNET ones) give large messages lower priority for transport than smaller ones so they take longer to be sent. And some electronic mail systems charge users by the character sent. All of the coding techniques described below tend to make coded files larger than the original WordPerfect ones. If you are concerned about size, it makes sense to "compress" the WordPerfect file before encoding it. Compression involves a different kind of file coding that reduces the space the file takes up on disk. This comes at the expense of convenience, since you must decompress it to use it. Popular compression tools include LHARC (otherwise known as LHA), a public domain tool, and PKZIP, a shareware one*.

Compression has another advantage. Popular compression programs provide automatic "integrity checks" to be sure that what you get out is the same as what you put in; the file encoders don't do as good a job of this. It you use a compression program that incorporates such checks, and the decompression process "passes", you can be reasonably assured that nothing unpleasant happened in the email transmission.

If you decide to compress, insert the following steps into the above in the obvious places:

* Tools of either type can be distributed to others, but shareware ones are not public domain: you are expected to register and pay for them if you use them. Registration is especially important with the current version of PKZIP, since registration gets you an excellent manual (the shareware documentation is a little thin) and several additional features. All public domain programs are free. Some free programs are not public domain but come with restrictions about copying, acknowledgments, or incorporation into other products.

Step S.2: Compress the WordPerfect file. If you are using PKZIP, and the WordPerfect file was called "mypaper.w51", you might type:

```
PKZIP MYPAPER MYPAPER.W51
```

This will create a new file, "mypaper.zip" that should be used in Step S.3 instead of "mypaper.w51".

Step R.4: If your correspondent used compression, then the next step after decoding will be to decompress the file. If the file was compressed with PKZIP, and, when you decoded the file, it was produced as "herpaper.zip", then you would decompress by typing:

PKUNZIP HERPAPER

to get "mypaper.w51" back. The file has now been decoded and decompressed and is ready for WordPerfect.

The Encoding and Decoding Programs

There are several different combinations of encoding and decoding programs available. The main criteria to be used in selecting them are

- You and your correspondent must have a matched pair.
- The encoding system used must be at least robust enough to survive the things that different programs and gateways do to electronic mail.

There are three sets in common use. They are *uuencode* and *uudecode,* which are used heavily in the UNIX world; a format called *BOO*, developed at Columbia University as part of the MSDOS Kermit package; and a new format called *base64*, developed as part of the recent email extensions efforts. The "seven-bit transfer encoding" format supplied with WordPerfect, and designed for the same purpose as the more common tools, has not proven to be robust over wide area network connections.

uuencode. While these programs were originally specified and developed for UNIX, there are several implementations of them for MSDOS. The encoder is usually called "uuencode", the decoder "uudecode". While widely used because it comes with most versions of UNIX, uuencoded files will not survive passage in either direction through most Internet-BITNET gateways. Files that are damaged in this way either will not decode at all, or will produce error messages or damaged data when read into WordPerfect. Consequently, this package should be used with care, and perhaps only when you understand the network path between your computer and that of your correspondent.

Syntax: uuencode WordPerfect-file Encoded-file

We suggest that the encoded-file be given the same name as the WordPerfect one, with the extension suffix ".UUE". That extension is a nearly universal convention.

Syntax: uudecode Encoded-file.UUE WordPerfect-file.W51

BOO. These programs were developed, as mentioned above, as a distribution form for MSDOS Kermit, but can be used for any "binary" file including WordPerfect ones. The characters and structure used work well through the Internet-BITNET gateways, within each network, and generally worldwide.

The only known limitation is for Latin America, where some local hosts use a character set translation that will damage these files. The encoder is called "msbmkb" (MSKermit-Boo-MaKe-Boo) and the decoder is called "msbpct" (MSKermit-Boo-PC-Translate).

Syntax: msbmkb WordPerfect-file.W51 Encoded-file.BOO

We suggest that the encoded-file be given the same name as the WordPerfect one, with the extension ".BOO". The use of that extension is a nearly-universal convention.

Syntax: msbpct Encoded-file.BOO (The name of the input file is stored in the BOO file.)

One additional advantage of the BOO programs is that a version of the decoder is available as a BASIC program in ASCII form. Virtually all versions of MSDOS come with BASIC. When your correspondent does not already have decoding programs, you can email the decoder in BASIC form in one message and send the (faster and more robust) decoder and encoder in BOO form in other messages. Your correspondent can then use the BASIC program to decode MSBPCT.BOO into MSBPCT.EXE. It can then be used to decode MSBMKB.BOO, at which point you are ready to start transferring WordPerfect files.

Base64. This form is relatively new, but has been designed to be even more robust than BOO. It provides the primary encoding used in the new MIME mail formats (see below). Several MSDOS implementations are now being tested; most should be public domain.

Some Additional Hints

Many people who use their MSDOS machines almost exclusively to run WordPerfect and similar programs don't have a systematic way of naming files. When one starts to have many versions of the same file, e.g., a WordPerfect version, a compressed version, and an encoded version, it can become difficult to keep them straight. We recommend that the suffixes (or "extensions") on files be used to distinguish the format of those files, e.g., always naming WordPerfect 5.1 files to end in ".W51", ZIP files in ".ZIP", BOO files in ".BOO", uuencoded files in ".UUE", and so on. The examples above reflect this approach.

BITNET Hosts and Internet Hosts

The discussion above makes the distinction between "BITNET" and "Internet" hosts. It has been traditional among users in much of the academic and research community to ignore the differences, and people are often confused about their own hosts and mail systems. To an increasing degree, hosts that have traditionally been connected to BITNET only have acquired Internet connections as well, some institutions have dropped out of BITNET entirely. When sending ordinary mail, the distinction is not important as long as you have a valid address: mail can be sent between BITNET hosts using an eight-or-fewer character host name; mail from BITNET hosts to Internet ones, and between Internet ones, requires a "fully qualifier domain name", usually something like INFOODS.UNU.EDU or Crop.Palm.CRI.NZ or NALUSDA.GOV. The distinction does become important when transferring coded messages or special files between system when every character and the file format are important, since passing messages between BITNET and the Internet inevitably involves some character set translations. If you know what is being used, you may be able to take advantage of it; if you don't it is better to be as careful as possible.

The Future

The electronic mail extensions efforts mentioned above include new formats for structured messages, including messages that include pictures, sound, and "applications" files, such as those of WordPerfect. These changes (most of which are known generically as "MIME") are being rapidly deployed on Internet hosts and some readers of this paper may already have access to them. For technical and stylistic reasons, things are happening somewhat more slowly in the BITNET environment, but the changes will penetrate there as well. When you and your correspondents both have access to MIME-aware mail systems, it should be possible to simply upload a WordPerfect file to the computer from which you send mail, identify the file as having come from WordPerfect in a particular version, and mail it. The mail system will then take care of everything else, including base64-encoding of the file being transmitted and decoding at the recipient end. The other importance of having base64 decoders available involves being able to extract information from a received MIME message even when your machine does not yet support MIME. Perhaps many of us will be using those tools by next year's conference.

Obtaining the Programs

The programs are available at the conference, by FTP and from an electronic mail server at INFOODS.UNU.EDU, or by sending a diskette (any reasonable MSDOS format) with a stamped, self-addressed mailer to the INFOODS Secretariat at the address specified on the first page.