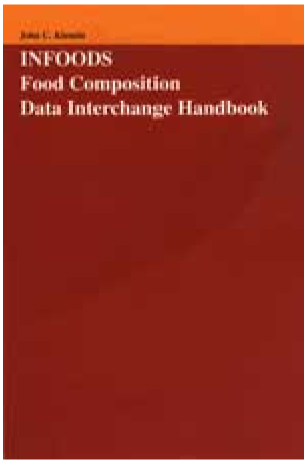

INFOODS Food Composition Data Interchange Handbook

JOHN C. KLENSIN

United Nations University Press

The views expressed in this publication are those of the author and do not necessarily reflect the views of the United Nations University.

United Nations University Press
The United Nations University
53-70 Jingumae 5-chome, Shibuya-ku
Tokyo 150, Japan
Tel.: (03) 3499-2811. Fax: (03) 3406-7345.
Telex: J25442. Cable: UNATUNIV TOKYO.

Printed in Hong Kong

WHTR-16/UNUP-714
ISBN 92-808 0774-9
United Nations Sales No. E.91.IIIA.6
03000 P

John C. Klensin is Principal Research Scientist in the Department of Architecture, Project Coordinator for INFOODS, and Director of the INFOODS Secretariat at the Massachusetts Institute of Technology, Cambridge, Massachusetts.

Contents

Acknowledgments p. 3

Part I: Introduction and overview p. 5

1. Introduction to the Interchange System
2. Technical overview
3. Introduction to the reference material

Part II: The reference sections p. 26

4. The header elements
5. The food element and subelements
6. Data values and data description

Part III: Processing data and interchange files p. 115

7. Registering elements
8. Conversion of data to interchange format
9. Conversion of data from interchange format

Appendix A registered international food record identifiers p. 138

Appendix B element registration form p. 139

Glossary p. 143

Bibliography p. 149

Acknowledgments

The United Nations University is an organ of the United Nations established by the General Assembly in 1972 to be an international community of scholars engaged in research, advanced training, and the dissemination of knowledge related to the pressing global problems of human survival, development, and welfare. Its activities focus mainly on peace and conflict resolution, development in a changing world, and science and technology in relation to human welfare. The University operates through a worldwide network of research and post-graduate training centres, with its planning and coordinating headquarters in Tokyo, Japan.

The International Food Data Systems Project (INFOODS) is a comprehensive effort, begun within the United Nations University's Food and Nutrition Programme, to improve data on the nutrient composition of foods from all parts of the world, with the goal of ensuring that eventually adequate and reliable data can be obtained and interpreted properly worldwide. At present in many cases such data do not exist or are incomplete, incompatible, and inaccessible.

This volume is the fourth in a series that provides information and guidelines about requirements for food composition data, the identification of nutrient and non-nutrient components of foods, the computer representation and accurate interchange of food composition data, and on the organization, compilation, and content of food composition tables and data bases. It presents the structure and rules for moving data files between countries and regional organizations in a way that preserves all of the information available. The approach also alerts the developers of data bases about potential areas in which ambiguities are likely and special care should be taken and identifies some mechanisms for improvement of overall nutrient data base quality.

Many people made significant contributions to the development of the INFOODS Interchange System. In particular, Dr. David Peterson and Ms. Roselyn M. Romberg contributed many ideas and some text to earlier versions of this document. Comments from Mr. Craig Franklin and Dr. Zita Wenzel, as well as Dr. Peterson, helped to determine the special forms of using the SGML Standard. Professor Vernon R. Young, Dr. Lenore Arab-Kohlmeier, Ms. Diane Feskanich, and several others provided feedback at critical times about the relationship of the evolving system to possible practice with nutritional data and their use. Anders Møller, Lena Bergström, Brucy Gray, Pam Verdier, and the New Zealand Division of Scientific and Industrial Research provided data against which the conversion models could be effectively tested, some of which is incorporated, with permission, in the examples. The material on the description of data files being interchanged and on the description and classification of foods is a formalization of material, some of it still unpublished, developed by the INFOODS Committee on Terminology and Nomenclature, headed by Professor Stewart Truswell. The data description section derives from several discussions and position papers about the basic character of ideal descriptive statistics for small samples and unknown distributions with Dr. Ree Dawson and Professor William M Rand, both of whom also made frequent and helpful comments about other parts of the manuscript and the working documents that were the foundation for it. Finally, the system was presented in technical detail at a special Oceaniafoods technical workshop on food composition data base organization and interchange. The participants in that workshop provided invaluable feedback on both technical issues and on the clarity of some of the concepts.

Any work of this type is ultimately a synthesis of many ideas and concepts. Much of the credit should go to those who contributed; the blame for the interpretations rests, as always, with the author.

Part I: Introduction and overview

1. Introduction to the Interchange System

AN INTERCHANGE SYSTEM FOR FOOD COMPOSITION DATA

A major goal of INFOODS has been the development of easy and accurate interchange of food composition data among countries and regional organizations. Such data exchange will obviate the perceived need for a single international data centre which holds all of the world's data, replacing it with distributed arrangements in which most data are held by their compilers, or by regional data centres operated by organizations of which data compilers or owners are members, until the data are actually needed.

It is not sufficient merely to move data back and forth. Food composition data are complex and often are, or should be, accompanied by extensive description of the foods being reported upon and the methods of analysis used. It has become clear in the last few years that the introductory material in a printed table may be nearly as important as the data values (see, for example, Arab et al. [1]). The need for such description and explanation arises through the necessity of comparing data from widely differing cultures. Not all food composition tables and data bases have the same level of description, however, and the informal text of an introduction is not the best way to communicate the information that is available, especially if it is to be processed automatically (e.g., by a computer), rather than simply read by trained scientists.

Other distinctions have been noted about various types of tables and data bases. Some data bases are oriented toward end users, others for national reference purposes, and still others are the fundamental collections of laboratory-level data before aggregation [24]. An effective interchange mechanism must be able to handle any of these types of data, without obscuring the differences in the types of information contained in each.

As one examines international data interchange, it becomes clear that the primary criterion for designing and evaluating a data interchange system is that it preserve whatever information actually is available, without forcing the data supplier to provide any more information than is known or imposing any more burden than is absolutely necessary. It would not be reasonable to try to require data suppliers to supply information which they do not know, or do not normally keep for their own purposes. Similarly, while in an ideal situation everyone might do things in the same way, the interchange system must be able to accommodate methods of reporting and data organization that some scientists might consider inappropriate. The inclusion of a way of expressing a particular concept in this document is therefore not necessarily a recommendation of that concept. Indeed, in a few cases, the text recommends against styles of data presentation and identification for which provisions are nonetheless made. Because identical and accurate sampling and analytic procedures, food selection, data description, and reporting are unlikely to ever occur in all tables, successful and meaningful exchange of food composition data has necessitated developing new conventions and technologies to organize and identify the many and varied components of these data.

Accurate comparison of data values requires very precise identification of how the values were derived and what they mean. When existing food composition tables and data bases are considered without their sometimes detailed introductions and appendices, there are often major ambiguities concerning the exact identification of foods, nutrients, units, and analytic and sampling methods. More careful comparison of food composition tables shows that different provide information about different nutrients, different types of foods, and different amounts and types of supporting information about samples, quality, recipes, and so on.

While any approach must accommodate the data that exist, the nutrient composition field continues to evolve. New food coding systems are introduced frequently, and changing hypotheses about the relationship between foods and health result in the introduction of nutrients that were not previously considered interesting into tables and data bases. If an interchange arrangement is to be useful for more than a few years, it must be "extensible", i.e., it must provide for new terminology, technology, and areas of interest to be defined and added to the system without compromising existing files and programs.

The differences in values and the ambiguities of data and food identification inherent in existing food composition data require that any interchange model operate on the assumption that actual tables and data bases cannot be expected to conform to a single standard or format. The interchange strategy must be descriptive of what decisions have been made about foods, food classification, nutrients, chemistry, or description and how those decisions have been carried out. At the same time, as suggested above, it cannot be dominated by norms about the "right" way to do things: even questionable data, poorly organized, may be more useful than no data at all, especially if the nature of the problems can be carefully identified and understood.

Partially as a result of the fact that particular data may be acceptable for some purposes and not for others, another goal of the interchange system is to permit tracing the flow of values, through copying (borrowing) or calculation, from one table to another and, more important, to be able to trace and assign responsibility for those values. All of the requirements for information that must be supplied with interchange files are the result of this tracking requirement.

To permit data interchange without loss of quality, and to encourage improvements in quality, data description, and data definition, INFOODS has designed a system of regional data centres and has developed an "interchange system" by which whatever data exist and are of interest can be transferred among regional centres with precise identification of values and without any loss of information. The interchange system is both a model of how data can be transported between regional centres and a data interchange format definition. As the latter, it is derived from principles of "generic markup" which are becoming increasingly important in the processing and exchange of textual documents. The standard for generic markup is specified in widely adopted international standards based on an International Organization for Standardization document, ISO 8879 [53]. Using generic markup has several special attractions, including its growing availability, the ability for people to directly inspect the format and content of the files, and the lack of dependence on any particular medium or data-transport arrangement. The other alternatives which are possible in principle were systematically eliminated as infeasible or too restrictive [55].

The interchange system will be used internationally, to facilitate exchanging data among countries and regions of the world. As with other INFOODS work, the interchange system uses existing international standards whenever possible, even when the invention of a nearly equivalent set of conventions specific to food composition data might result in short-term convenience or compactness. For example, provision is made for expressing food names in national languages and character sets where necessary, but only when consistent international standards for those character sets have been established.

THE REGIONAL DATA MODEL

While the details are not discussed in this manual, operating regional data centres, affiliated with INFOODS, are assumed as part of the interchange system. Those data centres act as a focus for food composition data base activity in their regions of the world and as the host for data interchange activity. When data are needed, for example, in most circumstances the user requiring the data would contact his or her own regional data centre, which would make arrangements to obtain them from a distant regional data centre, which might, in turn, obtain them from an organization within its region. The interchange mechanisms described in this manual are required only for use between regions. While they may be suitable for use between a regional data centre and data providers or users within its region, and may also be suitable for the ongoing storage of some reference or archival data bases, regions are free to work out their own arrangements for intra-regional communications and data interchange. A region that has specified its own data interchange formats and arrangements will presumably provide the capability to convert between the formats and conventions specified in this manual and its own formats at its regional data centre.

A regional data centre will typically be operated as part of an INFOODS regional liaison group, but this is not a requirement; either could exist independently of the other, and the term "regional data centre" is used instead of "regional centre" to stress this distinction. In principle, the regional data centre for a particular region need not even be located in that region, although it would usually be desirable for it to be.

In addition to acting as a focus for data interchange activities for its region, a regional data centre is expected to act as a registrar of international food record identifiers for the associated region, maintain current lists of interchange system tags and other identifiers, and keep records of tables and data bases originating in the region. It may also maintain some data locally, either from within the region (for easy export or as part of regional support functions) or from outside the region but frequently needed within it. In either of these cases, the regional data centre is expected to make special provision to ensure that its copies of data sets are kept up-to-date or that they are discarded when they are no longer current.

THE INTERCHANGE SYSTEM AS A CONCEPTUAL DATA BASE MODEL

While the principal design goal for the interchange system is information-preserving exchange of data among regional centres, its provisions for precise identification of nutrients and other food components, detailed recording of varying amounts of data about each nutrient and descriptions of those values, and ability to accommodate multiple coding, classification, and description systems may make it appropriate for national or regional use for archival and perhaps reference data bases. INFOODS has not made a specific recommendation that it should be used this way, but if the character of the data and description associated with a data base creates difficulties in using conventional data base systems with statistical or scientific data [4, 18, 27] the architecture of the interchange system, and software developed to handle it, might be considered as an alternative.

THE CONCEPT OF AUTHORITY

Food composition data, like most other scientific data, are rarely "true" or "false" in any absolute sense. Instead, the data values, the choice of foods, the decisions about whether two samples represent the same food, or a set of samples adequately represent some particular food, all represent scientific choices, not completely deterministic outcomes of perfect processes. In particular, it is possible, indeed likely, that different but equally skilled scientists would make different decisions, especially under different circumstances or assumptions about the user population and its needs.

As part of the important goal of preserving to the greatest extent possible all of the information about data being stored or transferred, an interchange system must move beyond traditional styles of exchanging only individual values in two important ways:

- It must encourage asking, answering, and documenting questions about what person or organization made a particular decision and will take responsibility for it. For example, in Chapter 4, a restriction is imposed that a single interchange file must have only one "source". This does not imply that all the data must come originally out of the same laboratory, or even the same country. Instead, it recognizes that the activity of putting together a data base involves editorial and scientific judgment, rather than mechanical concatenation of values. This is especially true when the data are derived from multiple sources. If nothing else, someone must conclude that combining the various values and considering the combination "one data base" makes sense. As soon as that decision is made, we have a new data base, containing new information-the decision itself-not just a combination of other data bases. And that implies a new, separate (and single) source.

Similar issues apply at the level of "individual foods". As discussed in Chapter 5, each collection of data associated with "a food" is associated with a food record identifier. A data base may contain multiple records for a given food, with different sets of values. If it does, each of these records will have a different food record identifier. The decision about whether a single food should have one or several food records is made by the table compiler. The interchange system imposes only two rules: (i) If previously published and identified data for an entire food (i.e., a single food record) are copied together, the food record identifier must be the same as the corresponding one in the original or data base. That is, the authority and responsibility for the integrity of the data rests primarily with the compiler of the original table or data base (but not the decision to include the data in the particular new data base). (ii) By contrast, if a food record is assembled from multiple sources-e.g., proximates and vitamins from one country and minerals from another- several key scientific decisions go into the compilation and combination process, and a new food record identifier is assigned to the newly created food record.

- Biological variability, variations in recipes, and many other factors contribute to there rarely, if ever, being a simple, firm value for the amount of any component in any food. Instead, the values typically represent some estimate of a particular parameter or other property of a statistical distribution. The interchange system provides extensive mechanisms for describing the distributions, and knowledge of and beliefs about them, in addition to simple values, or values and standard deviations or errors of estimate heretofore prevalent. These facilities are discussed in Chapter 6. While significant use of these facilities is not anticipated during the first years of data interchange, it is intended that they should provide a model for structuring more detailed information. That information should gradually become available as sophistication increases about data management and reporting within the food composition data user and supplier communities.

THE ROLE OF THIS MANUAL

This manual defines the organizing principles and formats of the interchange system-the model by which data about food composition can be transferred from one facility (typically a regional centre) to another while structuring and preserving whatever information may be available. It also specifies the ways in which the interchange system and its elements can be extended to account for changes in scientific conventions or knowledge without requiring data bases to be changed or programs to be rewritten if the changes are not important relative to the content or users of those particular data bases or programs.

The interchange system, of which an overview appears in the next chapter, depends on these principles and on conventions about the syntax in which textual and numerical values are written. As with conventional textual use of generic markup, the essential syntax uses a collection of carefully-defined "elements" which, in turn, are identified by a collection of specifically-defined "generic identifiers". Generic identifiers are predefined word-like strings of characters used to distinguish one element type from another.

More precise definitions of these terms, and examples of how they apply, appear in the chapters that follow. Later chapters specify those elements which are part of the interchange structure itself; the structure of elements used to describe the origins of, and responsibility for, an interchange file; foods and the properties of data. While the structure of elements that contain data values about the quantities of individual components present in foods is specified here, the generic identifiers for the food components themselves are specified elsewhere, primarily in the food component identification listing [17]. The information in that book may be needed for in-depth understanding of some of the examples that appear here. With the exception of a few areas for which specific generic identifiers have not been assigned at the time of publication, every element that appears in this manual is described either in the reference seniors or in the food component identification listing.

The general model of the interchange system is applicable to a great deal of food-related data which are not yet defined for use with it. Decisions to limit the extent of what to define have been conditioned by finite resources, the focus of the initial INFOODS mandate, and lack of clarity either of the needs or of the appropriate solutions. When additional elements of these are needed, working papers that begin to explore their development will be commissioned. These as yet unneeded areas and definitions include the use of national character sets for other than names of food, listing of recipes for mixed dishes, listing of food economics values (e.g., food balance data or food prices), and listing of food components that are not normally considered nutrients (e.g., food additives and contaminants).

PURPOSE AND AUDIENCE

This manual provides sufficient information about the interchange system to permit programs to be correctly written that will produce and interpret interchange files. Readers who are only concerned about a general introduction to the interchange system should concentrate on Part I, reading quickly through the balance with the confidence that most of the details are not important to them. Nonetheless, this is a technical document, and some terminology is used in very precise ways. The glossary contains all such terms, and should be consulted when there is doubt about whether a word is being used casually or with some special meaning.

Finally, this manual does not discuss the particular methods of transporting an interchange file from one location to another. The interchange system is designed to be insensitive to the choice of media (e.g., magnetic tapes or floppy diskettes) or transport mechanisms (e.g., computer networks or the post), depending only on a specially-delimited "interchange file". Since an interchange file consists only of text, it can be transported by any medium-including file transfer or electronic mail in computer networks; magnetic or optical recording on tapes, disks, paper, or diskettes; or even such older media as punched cards or paper tape-so long as the medium is able to transport eight-bit characters accurately. If elements that can contain "national characters" are removed from the file before it is sent or ignored when it is received, transmission with media that can process only seven-bit characters, or even low-quality computer printouts and telefax transmission and subsequent optical scanning are feasible. The only requirement is that the interchange file must be clearly separable from other information, a requirement that the file definition itself enforces. Sender and receiver should, of course, reach agreement about the media and mechanisms to be used before data are actually transmitted. Conventions about media and mechanisms for interchange among INFOODS regional data centres will be developed depending on the facilities available at those centres.

2. Technical overview

BASIC TERMS AND DEFINITIONS

The structure of an interchange file is described in terms of elements, or precisely identified blocks of data. The element is the basic "building block" of an interchange file, and serves to identify and contain the actual data being exchanged. Elements provide a structure for the data which is logically ordered for machines and relatively easy to follow for human beings. A typical element might be:

<NA> 5 </NA>

Elements are identified by tags, which identify and surround contents. In the example above, <NA> and </NA> are the tags which surround the content "5". Some elements use only a single tag, and are delimited by the next tag in sequence, whatever it might be. For example:

<date> 1983.11.04

Here, the content is the string "1983.11.04" in ISO standard date format [41], meaning "4 November 1983", the actual data content of the element. Contents may be data values (i.e., numerals or unrestricted strings of text), keywords (i.e., special values from a restricted list), other elements, or a combination of values, keywords, and elements. Elements that occur within other elements are said to be subsidiary or nested, and the term immediate is used to denote direct nesting, without intermediate elements, when the distinction is important. The following example, a brief but typical food component or <comp> element, contains a combination of data values, keywords, and nested elements and illustrates these concepts:

```
<comp>
<VITC> 30 </VITC> <NA> 0.12 <unit/> MMOL </unit/> </NA>
</comp>
```

In this example, the food component element consists of two tags, <comp> and </comp>, called the start-tag and end-tag respectively, and a content of two nested elements. The first element is the vitamin C element, whose tags are <VITC> and </VITC> and whose content is the actual data value "30 milligrams per 100 grams edible portion of food" (the units are specified as the default in the definition of the tag associated with the identified food component [17]). The second subsidiary element is the sodium element, whose tags are <NA> and </NA> and whose content consists of a value and a subsidiary element which specifies the unit of measure. The unit element's tags are <unit/> and </unit/> and its content is the keyword "MMOL", which stands for "millimoles". The <VITC> and <NA> elements are immediately subsidiary to <comp> . <Unit/> is immediately subsidiary to <NA>, subsidiary (but not immediately subsidiary) to <comp>, and not subsidiary to <VITC> at all. When it is clear from context which is meant, as in the case above, the start-tag is referred to as if it were the element. For example, in the previous sentence it would be more precise to say "The <unit/> element is immediately subsidiary to the <NA> element...".

Spaces before and after elements and line breaks are ignored in the interchange system. Hence the example above could be written all on one line, or with the sodium and vitamin C elements on separate lines, and so forth.

STRUCTURE OF AN INTERCHANGE FILE

In order to permit processors for interchange files to interpret them accurately and efficiently, interchange files must adhere to certain structural conventions. Consistent structure for all interchange files facilitates ease of use and interpretation of the data, both by people and by machines.

Every interchange file contains a single <infoods 85> element. Other types of information, such as data about the transport medium (e.g., magnetic tape density), electronic mail headers, telex information, mailing addresses, and informal text associated with the transportation of the file may surround but are not part of an interchange file.

The <infoods 85> start-tag is the only tag in the interchange system which requires an "attribute" indicating the version of the interchange system in use, in this case the version dating from 1985. The first tag of an interchange file must appear, therefore, as <infoods 85> and the last one must be </infoods>.

The <infoods 85> element's content is made up of two or more subsidiary elements, appearing in this order:

- a <header> element
- an optional <dflt> element, and
- one or more <food> elements.

The <header> element identifies the seeder end the source of the date. The <dflt> element identifies defaults which apply to the entire data file, such as weights and measures. The <food> element classifies the specific food, identifies any relevant measures, and supplies the relevant nutrient composition data for the food. The structure of an interchange file is therefore:

<in foods 85>
<header>
source and sender elements
</header>
<dflt> *default elements* </dflt>
<food>
<classif>
<ifiri> *food record identifier* </ifri>
other classification elements
</classif>
<fddflt> *per-food default elements* </fddflt>
<comp> *food component data elements* </comp>
<drvd-comp> *derived food component elements* </drvd-comp>
</food>
other food elements, starting in <food> *and ending in* </food>
</infoods>

While the <header> element is supplied once and not repeated, and the <dflt> is either omitted or supplied once, the first <food> element would ordinarily be followed by additional <food> elements, since it would be rare to transmit information about only a single food. All interchange files must adhere to this structure as outlined in the example above. (Again, line breaks are ignored in actual interchange; they are used in this book merely to enhance readability.)

OVERVIEW OF THE INTERCHANGE FILE PRIMARY ELEMENTS AND ELEMENT GROUPS

The Header

The <header> element of an interchange file provides information about the sender of the file and the source of the data. This information is critical in identifying the data for interpretation and for archival and tracking purposes. The <header> element is composed of two subsidiary elements, the <sender> element and the <source> element, each of which is composed of a number of required elements with several additional elements optional. The list of <header> elements and their definitions is inspired by the work of the INFOODS Committee on Terminology [33].

The Sender Subsidiary Element

The <sender> element of the header is composed of elements that identify the sender of the interchange file. This is the person or organization responsible for preparing the file at hand for transmission, not the person or organization responsible for the data values. The information in this element must be available to the receiver or user of the file to permit contacting the right person if there are problems with the organization of the data.

Required elements include those for name, organization, address, location or country of sender, postal code, and date of transmission of the file. While some of the information is redundant, the repetitions are important for sorting and classification purposes. Optional elements include those for additional information which is useful but not critical, such as the sender's title, electronic mail address, international telephone numbers (voice and fax), telex number, and cable code.

The Source Subsidiary Element

The <source> element of the header is composed of elements which identify the source of the data-typically a table or data base and compiler-being transmitted in interchange form. Only one data source is allowed per interchange file. Possible data sources may include food tables and other publications, nutrient data bases, laboratories, and so on. Optional elements include the address of the analytic lab if the source is a laboratory, the publisher's address for a literature source, or the ISBN number for a book.

The idea of a "source" involves several issues about what foods should be reported, or used, as a single "table" entity. It is most easily understood by analogy to the concept of data for a single food. The realities of chemical analysis and laboratory measurement make it improbable that nutrient values for a single analysis will all be from the same individual food item (e.g., the same apple), nor would we expect values derived from a single apple to have any special merit. Instead, one samples, homogenizes, and combines items to construct a laboratory sample [11]. The decision as to which apples are representative of "apple" or even of a particular cultivar and set of growing conditions is a substantive scientific one, and the criteria of "sameness" are neither trivial nor obvious.

While the <sender> element describes the origins of the interchange file, the <source> element describes the origin of the data values themselves. Information provided with <source> might be used to obtain additional scientific information about the data; information provided with <sender> is useful for technical problems with the interchange itself. In addition, <source> is expected to contain the information needed to reference the data in a publication that uses them. By contrast, <sender> would provide information for an acknowledgement of someone who had been particularly helpful.

The following is a complete sample <header> element:

```
<header>
<sender> <date> 1988.06.07
<fullname> Dr. J. D. Smith <fsnm> Smith
<orgz> EUROFOODS Regional Centre <addr/> Department of Human Nutrition <->
Agricultural University <-> De Dreijen 12 <->6703 BC Wageningen <-> The Netherlands
</addr/>
<country> NL <postcode> 6703 BC
<title/> Coordinator of the Laboratory </title/>
<phone/> +31 83 70 8 25 89 </phone/>
<telex/> NL 45015 </telex/>
</sender>
<source>
<ref/> Souci, S.W., W. Fachmann, H. Kraut. Food Composition and Nutrition Tables,
1986/87. Stuttgart: Wissenschaftliche Verlagsgesellschaft mbH, 1986.
<pub/> Wissenschaftliche Verlagsgesellschaft mbH </pub/>
<isbn> 3-8047-0833-1 </ref/>
<addr/>
Postfach 40 <-> D-7000 Stuttgart 1 <-> Deutschland
</addr/>
<country> DE <postcode> D-7000
</source> </header>
```

The above illustrates the combination of elements that do and ones that do not require end-tags and elements nested within other elements. The special tag <-> is discussed under "Repeated and Counted Elements" starting on.

Defaults

Default values for each component, such as the unit of measurement expressed per 100 grams of edible portion of the food, are included in the definition of the food component element [17], which is part of its registration. Default values which apply to data in the entire interchange file are specified in the <dflt> element. Subsidiary elements to <dflt> must reflect the structure of the food component or per-food default to which they refer. For example, if the data values for total protein, calculated from total nitrogen, for every food in the file were calculated using the standard conversion factor of 6.25, the <dflt> element for the file would look like this:

<dflt> <comp> <PROCNT> * STD </PROCNT> </comp> </dflt>

<Comp> and </comp> appear here because the <dflt> is treated as occurring at the same level as the <food> element itself. Hence, <comp> must be used to indicate that the subsidiary information applies to the specific food components.

The "STD" indicates that the standard conversion factor was used for all values ("-") supplied for total protein, calculated from total nitrogen, in the file. See the definition of <procnt> for more information.

The <dflt> tag itself acts as a "macro", affecting the interpretation of food component information. The rules by which it is applied are discussed in Chapter 3. Unlike <header> (and <sender> and <source>), <dflt> is optional and need not be supplied. If there are no default values, the element is omitted entirely.

The Foods

A <food> element contains the necessary classification information to properly identify a food, along with optional indicators of standard measures or other per-food defaults, followed by the actual nutrient data for that food. A <food> element consists of a maximum of four subsidiary elements:

- a required <classif> element,
- an optional <fddflt> element,
- either a <comp> element, or
- a <drvd-comp> element, or both,

where <classif> consists of information that identifies the data records and describes the food, <fddflt> identifies per-food defaults, <comp> contains the food component data (optional, but generally supplied), and <drvd-comp> contains the derived component data (optional, but often supplied depending on available data).

Classification Subsidiary Element

The <classif> element consists of the international food record identifier element, which is required, and any other classification elements necessary to identify the food for which data is provided. A very simple <classif> element that did not contain any food coding or classification information might look like this:

<classif> <ifri> ER.UK.M-W78.171 </ifri>
<bvname> Eggs poached </bvname> </classif>

In this example, the <classif> element is composed of the international food record identifier <ifri> element whose content identifies the food as that from the table classified as "EUROFOODS, United Kingdom, McCance and Widdowson 1978, Food Number 171", in this case, "Eggs, poached". The use of the <bvname> element indicates that the name is expressed in the ISO 646 basic character set.

Per-Food Default Subsidiary Elements

Default values which apply to data supplied for a single food may be specified in the <fddflt> element. For example, the <meas/> element supplies a denominator for data according to some common or standard measure. Such data may be provided instead of, or in addition to, data supplied according to the default measures registered for each nutrient. For example, in

```
<food>
<classif>
<ifri> NOA.USDA.HB8 4-78.09003 </ifri>
<FDA-FFV-8707> A143 B1245 C167 E150 F03 H003 J003 K03 M003 N03 P24 </FDA-FFV-8707>
<EUROCODE2> 10303 </EUROCODE2>
<bvname> apple, raw, with skin </bvname>
<bvname> malus sylvestris </bvname>
</classif>
<fddflt> <meas/> <-> piece <qty/> 150 </qty/>
<refuse/> 12 <cmt/> approx 8%; core and seeds considered inedible </cmt/> </refuse/>
<cmt/> approximately 3 per pound, 2.75 inch diameter </cmt/> </meas/>
</fddflt>
<comp> <NA> 3 <-> 7 </NA> </comp> </food>
```

sodium data is supplied first according to the defaults registered for the <NA> element, milligrams per 100 grams edible portion, and then according to the common measure, in this case, "piece", i.e., "per apple". The special character sequence "<-> " is used to separate multiple sets of values for the nutrient. It must appear with <fddflt> <meas/> as well as subsidiary to <NA> in order to specify that the special measurement applies to the second set of values, rather than the first set.

<Fddflt> is similar to <dflt> in that it is essentially an abbreviation indicator or macro with a specific range of applications. See the next chapter for details.

Food Component and Derived Food Component Subsidiary Elements

The <comp> and <drvd-comp> elements are composed of the elements for the distinct nutritive and non-nutritive components of the food. These typically consist of elements containing values (expressed in default units for the component) and, in certain cases, specific keywords from restricted lists to further identify or qualify the values or methods expressed. The initial set of generic identifiers subsidiary to <comp> and <drvd-comp> is specified in *Identification of Food Components for INFOODS Data Interchange* [17], and more may be registered as needed. A simple set of <comp> and <drvd-comp> elements might look like this:

```
<comp> <FE> 2.0 </FE> </comp>
<drvd-comp> <CHEMSC> 0.58 FAO73 </CHEMSC> </drvd-comp>
```

In this example, two data values are supplied for the food in question. The first value is for total iron, 2 milligrams per 100 grams edible portion [17]; the second value is for chemical score, 58% calculated using the 1973 FAO reference protein pattern [17].

DESCRIPTION OF THE DATA THEMSELVES

Most food composition tables contain, in addition to point values for each nutrient, some statistical description-typically a number of samples and a standard deviation or standard error-for them. While most tables contain mean values, other statistics about location are occasionally supplied. The requirement that the interchange system support the representation of any data that are available implies that it must be able to include any statistics that are available, and what those statistics mean. Statistical description of data is particularly important in interchange, where the receiver of a data file may need to assess the value for use in an unanticipated application or context, e.g., copying into a food data base for another country or imputing values for a similar food. A collection of optional elements are available for identifying which statistics are being reported and precisely identifying those statistics. Definitions of those elements, accompanied by an extensive discussion of the issues surrounding them.

SUMMARY

The Interchange System provides a format for each item of data required for the successful exchange of food composition information. Using the Interchange System's format of elements with uniquely assigned tags, an interchange file is readily interpretable both by machines and by people.

Explanations of semantic and syntactic conventions and detailed discussions of elements are found in the next four chapters.

3. Introduction to the reference material

This chapter provides information about the conventions used in Part II and about the principles for constructing interchange files that are not specific to any particular element or class of elements.

CHARACTER SETS

The text strings of which an interchange file is composed are, with a few exceptions, restricted to contain only a minimal set of characters. This permits these files to be displayed or printed on a wide range of devices in many countries. The characters are the graphics (plus "space") of the ISO 646 basic character set [40]. For a few specific situations, such as expressing the name of a food in a language that does not use the Roman alphabet, special provisions are made to identify the language, the alphabet, and the way the alphabet is encoded. Those provisions are discussed below.

Character Restrictions within Ordinary Data

The "less-than" sign (<) and the "greater-than" sign (>) are reserved for the construction and recognition of tags and may generally not appear within data. Thus, when reading data normally, any occurrence of "<" indicates the beginning of a tag; similarly, when reading data backward, any occurrence of "> " indicates the ending of a preceding tag.

Only a very small number of elements may contain data including "<" and "> ", and these are not permitted to have subsidiary elements. <Cmt/> elements-comments, which can have almost arbitrary character strings within them-and <ad/> and <x400/> elements subsidiary to <email/>-electronic mail addresses, which may require having the "greater-than" and "less-than" characters as part of the address-are the only elements of this type defined at present. The only strings the contents of these elements cannot include are their own end-tags ("</cmt/>", "</ad/>", and "</x400/>" respectively).

The space character ( ) plays a special role within formatted data. Line breaks (which may be system-dependent) and the tab character may also be used. Multiple spaces, tabs, and line breaks in this special role are treated as if only one space appeared; we use the term "whitespace" to refer to any sequence of consecutive spaces, tabs, and line breaks. The special uses of whitespace are discussed below.

Character Restrictions within Tags

A tag (except for <infoods 85>) consists of a generic identifier preceded by "<" or "</" and followed by "> ". Thus, a generic identifier may not include the " <" or "> " symbols, and must not start with the slant (/) It also must adhere to a number of other restrictions, listed below, to ensure that tags will have the same appearance, and the same meaning, regardless of printing device. These restrictions also help to prevent confusion between tags and actual data in an element, and identify some start-tags whose end-tag is required.

Generic identifiers must use an even more restricted subset of the ISO 646 basic character set than data. This subset consists of the numerals and letters (alphabetic characters), the hyphen, and the slant (/) All other characters, including the underscore (_), period, and space, are excluded. In addition, the slant may appear only as the last character of a generic identifier, the first character must be a letter, and hyphens must not appear adjacent to each other.

No distinction is made between upper- and lower-case characters in generic identifiers and keywords; i.e., <source> and <SOURCE> have the same interpretation. In unformatted text, there may be distinctions on the basis of case, as specified by the definition of the individual element.

Alternative Character Set Conventions

Exceptions to the very conservative ISO 646 basic character set are permitted for data values in a few elements. For example, an alternative character set may be used to spell out the local name of a food in its appropriate language. In such cases, the character set must be identified by the number of an ISO standard or the ISO registration number for that character set. The syntax for specifying an alternative character set is included in the description of the elements for which such characters are permitted.

CONVENTIONS FOR CONSTRUCTING ELEMENTS

Each element consists of a start-tag, content, and perhaps, depending on the particular element, an end-tag. The content consists of data, one or more subsidiary elements, or data followed by one or more subsidiary elements. Elements with no content are not permitted. For a discussion of the overall structure of an interchange file see the previous chapter.

Tags

A start-tag begins with " <" followed by an alphabetic character, while an end-tag begins with " </". Both end with "> ". Between the opening " <" or " </" and the closing "> " is a "word", the generic identifier. A generic identifier is constructed according to the rules under "Character Restrictions within Tags", above.

Some examples of tags are: <header>, </source>, <unit/>, and </unit/>. The corresponding generic identifiers are "header", "source", "unit/", and "unit/" (not "unit" or "/unit/"). The following are character strings that cannot be generic identifiers:

"ONE KIND", "3rd", "this&that", and "ANOTHER/TAG"

The special tag <-> is neither a start-tag nor an end-tag, though in certain cases it may act as both (see the section titled "Repeated and Counted Elements", below).

Formatted and Unformatted Data and Whitespace

Data can be formatted or unformatted. Formatted data consists of one or more numerals and/or keywords separated by whitespace (spaces, tabs, and/or new lines) whereas unformatted data is arbitrary text.

A numeral is a string of digits with an optional sign and/or decimal point, or a numeral in scientific notation as prescribed in the applicable standards [36]. Forms beginning with a decimal point should not be used, e.g., "0.4" should be used rather than ".4". A keyword has the same internal structure as a generic identifier except that it cannot end in a slash: it must start with a letter, and continue with letters, digits, and/or hyphens. For example, "0.128" is a numeral and "USDA" is a keyword. "0.128 USDA" is formatted data consisting of a numeral followed by a keyword separated by required whitespace.

A raw data string consists of either formatted data (one or more data values) or one unformatted data item, or both; if both, the formatted data must come first. In general, one cannot determine whether data are formatted or unformatted by looking at them; the definition of the tag and its content is required. Any formatted data, such as the example "0.128 USDA" above, could also be interpreted as an unformatted data item. On the other hand, "0.128USDA" can only be unformatted data: it is neither a numeral, because it contains letters, nor a keyword, because it starts with a digit.

Whitespace is required to separate successive formatted data items, and to separate formatted data from immediately following unformatted data. This whitespace is not part of the data item. Data items never begin or end with whitespace, although an unformatted data item may have embedded whitespace. For example, the string " This is a sample unformatted data value. " includes an unformatted data value consisting of 41 characters beginning with "T" and ending with ".". It has both leading and trailing whitespace, which are not part of-the data item. However, the spaces between "This" and "is", between "is" and "a", and so forth, are part of the data item.

Whitespace immediately before and after tags is ignored. This means that data always may have whitespace before or after. Optional and extra whitespace in the form of judicious indenting and line breaks can make the structure of an interchange file much easier for a person to read.

Contents

The content of an element consists of all of the characters between the start-tag and the end-tag of the element. The content of an element can be subsidiary elements or a raw data string, or both. If an element includes both raw data and subsidiary elements, the data must come first. Each type of element (as designated by its generic identifier) has a specific list of what data values and/or subsidiary elements are permitted or required within the content of that type of element.

No element has an empty content. If all of the subsidiary elements are optional and none are desired, then the element itself must also be optional and should be omitted; similarly, if it is to contain a data value and that value is non-existent, the element itself should be omitted.

In the following example, the content of the <VITB12> element is data, the numeral "03":

<VITB12> 03 </VITB12>

In the following example, the content of the <comp> element is two subsidiary elements. The first is the same <VITB12> element shown above, whose content is data. This subsidiary element is followed by a second subsidiary element, <VITE>, whose content consists of a data numeral followed by three subsidiary elements (<XBTP>, <XGTP>, and <XATT>), whose content is in each case a (data) numeral:

```
<comp>
<VITB12> 03 </VITB12>
<VITE> 0.7 <XBTP> 0.4 <XGTP> 0.1 <XATT> 0.26 </VITE>
</comp>
```

The only elements that do not require an end-tag are those that permit only a small number of formatted data items (numerals or keywords) or a single unformatted data item in their content. They do not permit subsidiary elements. These elements never have an end-tag; end-tags are never optional. Each such element is so identified as part of its registered description. For example, <VITE12> and <VITE> elements require an end-tag, but <XBTP>, <XGTP>, and <XATT> elements do not.

The Trailing Slash and End-tags

Whether or not an end-tag is required can be predicted from the form and type of the generic identifier. Conversely, the form of a generic identifier is determined by the context in which it is used and whether or not it requires an end-tag. Specifically

- All structural elements (see below) require end-tags and their generic identifiers do not end in slashes.
- All elements immediately subsidiary to structural elements require end-tags and their generic identifiers do not end in slashes. This category comprises subsidiary structural elements (also covered by the rule above) and elements immediately subsidiary to the "boundary" structural elements described below.
- Any other element that requires an end-tag has a generic identifier that ends in a slash. Elements that are not structural elements or immediately subsidiary to them and whose generic identifiers do not end in slashes do not require (or permit) end-end tags.tags.

These conventions are, admittedly, complex. From a conceptual standpoint, it would have been much easier to simply require end-tags for all elements. However, it became very clear in the early discussions from which the interchange system evolved that there was a critical requirement that small and simple data files should require minimal structural overhead so that, for example, they could be exchanged on low-capacity media (notably diskette) and processed successfully on small computers. Consequently, more complex rules were adopted that tend to keep small files small and impose more of the burdens of structure and precise identification on the files and data bases that would be proportionately larger and more complex in any case.

STRUCTURAL ELEMENTS

The <infoods 85> element and those elements that appear for a few levels of elements and content below it are used primarily to structure, i.e., to organize and order, the interchange file, rather than to carry table-specific or food-specific information. These are called structural elements. Structural elements always have end-tags, their generic identifiers do not end in slashes, and their content consists of elements only. Structural elements-except <infoods 85>-can occur only as subsidiary elements of other structural elements, and occur therein only in a prescribed order (although some are optional). In other words, a structural element may never appear subsidiary to a nonstructural element.

One of the implications of this is that some elements mark the nesting boundary between structural and non-structural elements: *no element subsidiary to them is structural, and all elements to which they are subsidiary are structural.* Those elements, which are themselves considered structural, are <header>, <classif>, <comp>, and <drvd-comp> .

The order in which the elements subsidiary to a given element must appear, if any, is always specified as part of the definition of the containing element. In general, the subsidiary elements must appear in a specific order. The major exception is for elements immediately subsidiary to the boundary elements listed above: those subsidiary elements may appear in any order.

OTHER ELEMENTS

Specific Food Component and Derived Component Elements

<*Specific component*> and <*specific derived component*> elements are the subsidiary elements of <comp> and <drvd-comp>, respectively. Like structural elements, they require an end-tag and their generic identifiers do not end in a slash. However, since they are not structural elements, they may occur in any order. (The terms *"<specific component>"* and *"<specific derived component>"* are shown in italics to remind the reader that they are not actual tags or elements but only placeholders for the registered list of identifying generic identifiers and element structures for food components [17].)

Other Non-structural Elements

While there are a few exceptions, other non-structural elements deal directly with data. Unlike structural or <*specific component*> or <*specific derived component*> elements, certain of these elements do not use end-tags. To avoid any question as to which do and which do not, each of these elements requires an end-tag if and only if its generic identifier ends in a slash.

For example, in the structure

```
comp <VITB12> 7 </VITB12>
<VITE> 3 <unit/> IU </unit/> c XATP> 1.0 <XBTP> 0.4 </VITE>
</comp>
<drvd-comp> <chemsc> 0.52 </chemsc> </drvd-comp>
```

an end-tag is required for the comp and <drvd-comp> elements because they are structural elements. The <VITB12> and <VITE> elements require end-tags because they represent specific food components and ate immediately subsidiary to the structural element <comp> . The <chemsc> element requires an end-tag because it is a specific derived component, subsidiary to the structural element <drvd-comp>. Each <unit/> element requites an end-tag because "unit/" ends in a slash. The <XATP> element, which is subsidiary to the specific food component <VITE>, does not take an end-tag, because it is not a structural element or immediately subsidiary to one and "XATP" does not end in a slash.

Element Values of "Zero", 'Trace", and "Missing"

If the value for an element is actually "missing", i.e., no value is available, the element is omitted entirely. This is a case of the principle that elements without content do not appeal. If a value, however suspect, is available, it should be included: even values of questionable accuracy may be useful to some users under some sets of circumstances. Statistical and data treatment elements should be used to describe and, if possible, quantify the uncertainties. Under no circumstances should a zero (or any other number) be provided for a missing value unless that is the table compiler's best estimate of the actual value, preferably identified as such.

When the food component is measured, a zero value can occur either as the result of there actually not being any of the component present or as the result of limitations of apparatus, instrumentation, or procedures. Especially in the case of an apparent measured zero, data description elements should be used to give the receiver information about the accuracy to which measurement could be achieved.

The presence of a small, but not accurately measurable, amount-the so-called "trace" amount-provides another situation in which the description of the data value provides more information than the value itself. The special data item "TR" may be used as a keyword in any situation in which a data value would otherwise appear, but it should be used only with sufficient data description to identify the circumstances under which the "trace" value occurred, e.g., with an explicit element that identifies the detection level for the method used.

A related but slightly different approach to these problems has been provided by Kent Stewart [30].

REPEATED AND COUNTED ELEMENTS

Most element types can occur at most once as subsidiary within a given element; a few can tee repealed. For example, the <infoods 85> clement can have only one <header> but may have many <food> elements. However, the various <food> elements are not distinguished by which their sequence: they are identified by internal data, not by the order in which they appear. Occasionally it is useful to have a repeatable element whose repetitions are distinguished by sequence. In this case, a very special notation is used. Instead of repeating the entire element, with the first end-tag adjacent to the next start-tag, the repeated contents are separated by a special tag, <->. For example:

```
<VITA> 13 <-> 7.2 </VITA>
```

where the first value would normally "per 100 g edible portion" and the second value would be for some common unit, such as "per piece". That unit would be specified in a previous <fddflt> element. If it were not, this notation would indicate that the food had values of both 13 and 7.2 micrograms pet 100 g edible portion, a contradiction (the choice of "micrograms" is part of the definition of the <VITA> element but could be overridden with a separate subsidiary element, <unit/>).

All specific food component elements (of which <VITA> is one) are of this type. On the other hand, the <addr/> element contains various lines which must be presented sequentially for the address to make sense:

<addr/> Post Office Box 1234
<-> Anywhere, Maine 00001
<-> USA
</addr/>

Only a very few element types (but including all specific food component elements) are permitted this ordered repetition mechanism. Each one that does is clearly specified in its registered description.

THE MACRO ELEMENTS <dflt> AND <<fddflt>>

Two special elements are also defined that can be used to reduce the size of files of data in interchange format or to reduce the complexity of creating such files. They are always optional, and while they may be very convenient for some producers of interchange files, others will find it best to ignore them. They do add complexity to the structure and processing of interchange files, and therefore probably should be omitted (or, as explained below, expanded before the file is sent) if small flies are being transferred in interchange format to users with limited computer expertise. INFOODS regional data centres are expected to have the capability of processing these elements.

The two elements are identified by the tags <dflt>, which appears immediately subsidiary to <infoods 85> (at the same level as <food>), and <fddflt>, which appears immediately subsidiary to <food> (at the same level as the <*specific component*> elements). <Dflt> is used to specify "default values" for all of the foods in the data base, while <fddflt> is used to specify "default values" for the components of a given food. Each has the same structure as the elements into which it substitutes; i.e., <dflt> has the same possible selection of content elements as <food>, and <fddflt> has the <*specific component*> elements as its content.

These elements are used as crude text "macros", providing for the substitution of values that do not appear directly in the content of <food> or <*specific component*> elements or their subsidiaries. The asterisk (*) is used to indicate the position of data that must be provided in the actual elements. To minimize processing complexity among these two elements and the elements to which their values are applied, there are no precedence rules: information may appear with <dflt>, with <fddflt>, or in the <food> element or its components, or not at all, *but not in more than one per category*. The element structure used in <dflt> or <fddflt> indicates where values are applied to the actual data elements. From a programming standpoint, the absence of precedence rules implies that a processing program can be constructed that will convert a file that contains <dflt> or <fddflt> elements into one that is fully expanded and in which they do not appear. With this model, the processing program requires no embedded knowledge of the specific foods or components. <Dflt> and <fddflt> may even appear together if they do not contain overlapping information. Such a program would continue to work with any future extension of the interchange system, including the addition of new elements. It could also operate independently of programs to convert or extract specific data from interchange files.

While other uses are possible-it can have any structure that <food> can have- <dflt> will typically be used to specify characteristics in common for all measurements of specific food components in a data base. For example, if all measurements of energy would normally be specified with the <energc> element with the "KJA" keyword, the following element could be provided:

<dflt> <comp> <energc> * KJA </energc> </comp> </dflt>

This would imply that any time an <energc> element appeared subsidiary to <food> and <comp> elements in the interchange file it would be treated as if "KJA" had appeared. In other words,

<food> ... <comp> ... <energc> 3 </energc> ... </comp> ... </food>

would be treated as if it read

<food> ... <comp> ... <energc> 3 KJA </energc> ... </comp> ... </food>

Because, as mentioned above, there are no precedence rules for substitution, the presence of the construction above would make it impossible to have any <energc> value in the file that contained a keyword specifying a method: *if different <energc> methods appear in the file then <dflt> may not be used to specify any of them.*

In this example, the content of <dflt> could also contain elements for other subsidiary elements of <comp>, for <drvd-comp> and its subsidiaries, and, in principle at least, for <classif> and its subsidiaries. At most, one <dflt> element is permitted in an interchange file.

The rules for application of fddflt are similar to those for <dflt>. If it appears, it applies to all the elements of <comp>, i.e., to all <*specific component*> elements. It will most often be used to express the units in which the food is reported, i.e., to provide the <meas/> element and its value for the entire food. Since the structure of fddflt parallels that of <*specific component*>*,* if the interchange file contains more than one set of measurements for each food component, the special delimiter element "<->" may be used to specify that the value of <fddflt> applies to only one. If <-> does not appear, it will be assumed to apply only to the first. So

<fddflt> * <-> <meas/> piece </meas> <fddflt>

would imply that, for any food components for which more than one value (or set of values, if full statistical information were provided: see Chapter 6) appeared, the second one would represent values reported "per piece".

No rule of the interchange system prevents using an <fddflt> element as a subsidiary of <dflt>. However, if this is done, the creator of the file must ensure that the food defaults apply to every food component in every food in the data base, and that no conflicts occur with values specified with the individual foods or components. In practice, the combination will be useful, if at all, only with highly specific data bases, e.g., ones reporting many measured values for the same food, as for different locations or seasons. In that situation, it might be sensible to provide <classy> and some of its elements as components of <dflt> as well.

Part II: The reference sections

4. The header elements

INTRODUCTION

This chapter contains descriptions of the interchange elements that serve to identify the interchange file and its origins. Each type of element will be introduced on a separate page with the description headed by the start-tag for the element. This is followed by a short general description, a description of the permissible content (format) for that type of element, and a discussion of the details, often with examples.

<infoods 85>

The <infoods 85> element is the overall structural element comprising an entire interchange file; i.e., its start-tag and end-tag identify the beginning and the ending of the interchange file.

Description

The start-tag includes the generic identifier (infoods), whitespace, and a two-digit year; both start-tag and end-tag are required. The content is an (immediately subsidiary) <header>, an optional <dflt>, and one or more <food> elements, in that order; this list is, in principle, extensible by registration. This element and its immediate subsidiaries are structural elements.

Format

An interchange file consists of precisely one <infoods 85> element. Its immediate subsidiary elements separate the collections of data about each food from each other and from the information about the source of the file and food data. The two-digit year (85) serves to identify this interchange format as distinct from any possible future revisions. Since the system is internally extensible, such revisions are not anticipated.

Example

<infoods 85>
<header>
subsidiary elements with information about the file source
</header>
<food>
subsidiary elements with information about a food
</food>
<food>
subsidiary elements with information about another food
</food>
additional <food> *elements*
</infoods>

In this particular example, there is no <dflt> element.

<header>

The <header> element is the first subsidiary element in the overall <infoods 85> element. In other words, it must appear immediately after <infoods 85> in an interchange file. It includes the information about the origins of the interchange file and the data therein.

Description

Both start-tag and end-tag are required. The content is a <sender> element followed by a <source> in that order, and both are required; this list is, in principle, extensible by registration. This element is a structural element, so its immediate (non-structural) subsidiaries all require end-tags and their generic identifiers do not end in slashes.

Format

The <header> identifies the sender of the interchange file and the source of the data within it. It has no immediate data. See <sender> and c source> for the details of the information to be included.

Example

<header>
<sender>
subsidiary elements with information about the person or organization transmitting the information
</sender>
<source>
subsidiary elements with information about the source of the food component data
</source>
</header>

<sender>

The <sender> element is the first immediate subsidiary element of the <header> structural element. It includes the information about the transmission of the interchange file.

Description

Both start-tag and end-tag are required. The content includes these required immediate subsidiaries (the numbers in parentheses are the page numbers on which their descriptions begin):

<date> (31)	<fsnm> (34)	<country> (39)
<fullname> (33)	<addr/> (38)	<postcode> (40)

and these optional (but see below) immediate subsidiaries:

<sendref> (32)	<title/> (41)	<telex/> (50)
<ianame/> (35)	<email/> (42)	<cable/> (51)
<orgz> (36)	<phone/> (48)	<cmt/> (52)
<contact/> (37)	<fax/> (49)	

These lists are extensible by registration as becomes necessary. There is no immediate data; the <sender> element's immediate subsidiaries are not structural elements, and so they may appear in any order.

Format

<Sender> contains data about the transmission of this interchange file: when it was created or transmitted (<date>), how it should be referred to when communicating with the sender (<sendref>), and who sent it (all the rest). The sender may be a person or a corporate entity; in either case, the sender is the entity identified in <fullname>. *The <sendref> clement specifically identifies this interchange file, not the data contained therein.*

If the sender is a person, then <orgz> is optional and normally there should be no <contact/>, i.e., the sender is the contact. If the sender is an organization, then there should be no <orgz>, but a <contact/> should be supplied. A person should always be identified either as sender or contact in case there are problems with automatic transmission which must be investigated. The rest of the locating information (e.g., telephone and telex numbers) in the remaining immediate subsidiary elements applies to the human sender or contact.

The <addr/> element contains a complete mailing address, including the name, title, postal code, and country, which is duplicated or expanded in separate adjacent elements. They are included in <addr/> to ensure that they are correctly located, punctuated, abbreviated, etc., within the address; they are also separate because a file recipient may be unsure of how to extract them from the mailing address correctly.

<date>

The <date> element is a required immediate subsidiary of the <sender>. It specifies the date the interchange file was prepared for transmission.

Description

The start-tag is required; there is no corresponding end-tag. The content of <date> consists of one unformatted data item that ends when another tag is encountered.

Format

The content of <date> is unformatted data which must consist of characters making up the date the interchange file is sent. Do not use dates of the form "1/2/88" or "1-2 88"; the conventions for indicating month-first versus day-first are not adequately well known nor observed. Use the internationally recognized convention "yyyy.mm.dd": it is rarely misused, is easy to read, introduces no problems at the turn of the century, and provides an easy-to-sort data item.

Example

<date> 1988.01.02

<sendref>

The <sendref> element is an optional immediate subsidiary of the <sender>. It specifies the way this interchange file should be referenced when communicating with the sender.

Description

The start-tag is required; there is no corresponding end-tag. The content of <sendref> consists of one unformatted data item that ends when another tag is encountered.

Format

The content of <sendref> is unformatted data which must consist of characters making up a name or phrase by which this particular interchange file can be identified in communications to the sender. It is optionally provided by the sender and is especially useful in identifying each of several interchange files being sent to the same receiver at about the same time. This might be needed by the receiver to describe which of several files had been correctly received and for the sender then to identify (by elimination) which files had been sent but not received.

Examples

<sendref> MIT/Harvard special data set 1
<sendref> NAregional.1988.10.19.0030

The second of these should not be mistaken for an international food record identifier. Although it looks somewhat like one, its use in this element indicates that it is a reference value, for the file rather than a particular food record, supplied by the sender.

<fullname>

The <fullname> element is a required immediate subsidiary of various elements. It specifies the complete name of a person or organization.

Description

The start-tag is required; there is no corresponding end-tag. The content of <fullname> consists of one unformatted data item that ends when another tag is encountered.

This element is usually used in conjunction with <fsnm>.

Format

The content of <fullname> is unformatted data which must consist of the characters of the name of the person or organization being named; if a person, it does not include any title. The names and initials of this individual may be given in any appropriate order, e.g., with the surname last for most of North America and Western Europe, with the family name first for Japan and China, and so on.

Unless the element appears as an immediate subsidiary of <ianame/>, this name must be transliterated, if necessary, into the characters of the restricted ISO 646 character set permitted for ordinary data in the interchange file. If the original name is normally written in characters that are not part of that set, it will often be desirable to provide that representation as part of an <ianame/> element.

Examples

<fullname> Joseph J. Smith
<fullname> Michele Gerard
<fullname> Massachusetts Institute of Technology
<fullname> Abdul Aziz

<fsnm>

The <fsnm> element is an immediate subsidiary of various elements, always in association with a <fullname> and is used for alphabetization and formal address. It specifies the name of a person or organization that is appropriate for sorting, retrieving, or formal address. "Fsnm" may be thought of as an abbreviation for "formal sort name".

Description

The start-tag is required; there is no corresponding end-tag. The content of a <fsnm> consists of one unformatted data item that ends when another tag is encountered.

Format

In general, any element having an immediately subsidiary <fsnm> element will also have an immediately subsidiary <fullname> element. The content of a <fsnm> is unformatted data which must consist of the characters of the name by which the person named in the associated <fullname> element (if it names a person) is addressed.

One purpose of a separate <fsnm> element, which duplicates information in the <fullname>, is to permit proper alphabetization independently of how the full name is presented in <fullname>. Hence, this field should also be specified for organizations, and will show all or part of the <fullname> in the appropriate order for alphabetizing.

The restrictions on the characters in <fsnm> are identical to those in <fullname>

Examples

<fullname> Joseph J. Smith <fsnm> Smith
<fullname> Michele Gerard <fsnm> Gerard
<fullname> Campbell Soup Company <fsnm> Campbell's
<fullname> Hasui Kawase <fsnm> Hasui

<ianame/>

The <ianame> element is an immediate subsidiary of various elements, and is used to designate the name of an individual or organization in the alphabet in which it is usually spelled where that alphabet is not a subset of the restricted ISO 646 alphabet discussed in Chapter 3. It will typically appear in conjunction with the conventional <fullname> and <fsnm> elements. "Ianame" may be thought of as an abbreviation for "international alphabet name".

Description

Both start-tag and end-tag are required. The content of <ianame/> consists of either a <fullname> element or an <fsnm> element, or both, followed by required <lang> and <charset> elements. The content of the <fullname> and <fsnm> elements, when used in this context, may be in any character set specified by the <lang and <charset> elements since they specify how the computer-readable characters are to be interpreted. One or more <cmt/> elements may be included.

Format

The content of <ianame/> consists of elements only; there is no immediate data. The subsidiary elements are <fullname>, but with characters in any defined language and alphabet in its content and <fsnm>, but with the same characters used for <fullname> followed by <lang> and <charset> elements. At least one of the elements <fullname> or <fsnm> must appear. <Lang> and <charset> are required, and are used as defined for <exname>.

Except in special circumstances, <ianame/> should not appear without <fullname> and <fsnm> at the same level; since many receivers of interchange files will be unable to completely process names in international alphabets, names should always be provided transliterated into the standard restricted alphabet as well as in their original alphabet.

Example

<sender>
<fullname> Agricultural Research Institute <fsnm> Agricultural
<ianame/> <fullname> Rannsóknastofnun Landbúnadarins
<lang> is <charset> 8859 1 </ianame/> </sender>

<orgz>

The <orgz> element is an optional immediate subsidiary of various elements. It specifies the organization to which the person named by an accompanying <fullname> belongs.

Description

The start-tag is required; there is no corresponding end-tag. The content of <orgz> consists of one unformatted data item that ends when another tag is encountered.

Format

<Orgz> should only occur as an immediate subsidiary of an element also having an immediately subsidiary <fullname>. The content of <orgz> is unformatted date which must consist of the name of the organization with which the person named in the corresponding <fullname> is associated. If the <fullname> names an organization, there should be no accompanying <orgz>.

Example

<orgz> University Food Composition Service

<contact/>

The <contact/> element is an optional immediate subsidiary of the <source> and <sender> elements. It specifies the person within an organization who acts as a data generator, compiler, or sender.

Description

Both start-tag and end-tag are required. The content of <contact/> identifies an individual, and normally consists of <fullname> and <fsnm> elements. If necessary, it may also contain any other elements, normally subsidiary to <sender> or <source>, that are needed to permit reaching this person efficiently: <addr/>, <country>, <postcode>, <title/>, <email/>, <phone/>, <fax/>, <telex/>, <cable/>, or <cmt/>. Normally these elements should not be repeated if the ones supplied with <sender> or <source> are adequate.

Format

<Contact/> should appear when the immediate content of <sender> or <source> identifies an organization, not a person. The content consists entirely of elements; there is no immediate data.

Example

```
<sender>
<date> 1990.07.04
<fullname> INFOODS Secretariat, Massachusetts Institute of Technology
<fsnm> INFOODS
<addr/> Room N52-457 <-> MIT <-> 77 Massachusetts Ave <-> Cambridge, MA 02139 <>
USA </addr/>
<country> US <postcode> 02139
<phone/> +1 617 253 8004 </phone/>
<contact/>
<fullname> John C. Klensin <fsnm> Klensin
<phone/> + 1 617 253 1355 </phone/>
<email/> Klensin@MIT.EDU <net/> INET <cmt/> From BITNET/EARN also </cmt/>
</net/> </email/> </contact/>
</sender>
```

<addr/>

The <addr/> element is a required immediate subsidiary of various elements. It includes all of the lines of the sender's mailing address.

Description

Both start-tag and end-tag are required. Successive "lines" of <addr/> are separated by the special tag <->. Each of these "lines", which need not be on separate lines of the interchange fife, consists of one unformatted data item. <Addr/> may also contain a <cmt/> element.

Format

<Addr/> is an element whose content is successive lines of a sender's mailing address, separated by the special tag <->. They must be in the proper order for use as an address; some of the lines presumably will duplicate information in the <fullname>, <orgz>, <country>, and/or <postcode> elements, which are included separately for sorting convenience and other purposes.

Example

```
<addr/> Dr. J. J. Smith
<-> Post Office Box 1234
<-> Anywhere, Maine 00001
<-> USA
</addr/>
```

<country>

The <country> element is an immediate subsidiary element of various elements. If associated with <addr/>, it specifies the country component of that address (the country name must still be included in the <addr/> element; specifying it separately is useful for sorting and source identification). It is required in addition to <addr/> in some of the contexts (including subsidiary to <sender>) where the <addr/> element is required.

Description

The start-tag is required; there is no corresponding end-tag. The content of <country> consists of either a keyword or an asterisk followed by one unformatted data item. The element ends when another tag is encountered.

Format

The content of <country> is a keyword consisting of the ISO 3166 upper case two-letter ("Alpha-2") code for the country for which the associated <addr/> is intended. It is provided as a separate field to permit easy sorting and extracting by country. If ISO 3166 does not define an appropriate two-letter code, the content of <country> consists of the asterisk "keyword" (*) followed by an unformatted data item, the complete country name-expressed in the restricted ISO 646 character set generally permitted for interchange file data. The two-letter code is to be used when it exists, as it does not have alternative spellings; this facilitates sorting and retrieval.

A current list of ISO 3166-associated country codes is available from the Secretariat. The list does change between official revisions of the Standard, so the Secretariat should be consulted if a country is not found in it.

Examples

<country> US
<country> DE
<country> TZ
<country> FJ

<postcode>

The <postcode> element is an immediate subsidiary of various elements. It specifies a postal code associated with an accompanying <addr/>. As with <country> (q.v.), it provides information that is deliberately redundant with that in <addr/> and is required in some of the same contexts in which the <addr/> element is required.

Description

The start-tag is required; there is no corresponding end-tag. The content of <postcode> consists of one unformatted data item that ends when another tag is encountered.

Format

<Postcode> always occurs as an immediate subsidiary of an element also having an immediately subsidiary <addr/>. The content of <postcode> consists of data characters giving the regional postal code for the associated address, in the format prescribed by that country's postal system.

Examples

```
<postcode> D-1000
<postcode> NG7 2RD
<postcode> 73170
<postcode> 150
```

<title/>

The <title/> element is an optional immediate subsidiary of various elements, associated with <fullname>. It specifies the professional title of an individual.

Description

Both start-tag and end-tag are required. The content of <title/> consists of one unformatted data item and an optional <cmt/> element.

Format

The content of <title/> is unformatted data which must be the professional title of the sender. If the sender has more than one professional title, it should be the one most relevant to that person's relationship to the interchange file or the data therein. However, compound titles are permitted when appropriate.

Examples

<title/> Professor of Nutrition </title/>
<title/> Professor of Chemistry and Director of the Analysis Laboratory </title/>
<title/> Director, INFOODS Secretariat <cmt/> Also Principal Research Scientist, Department of Architecture, MIT </cmt/> </title/>

<email/>

The <email/> element is an optional, repeatable immediate subsidiary element of the <sender> element. It specifies the sender's electronic mail address. "Email" can be interpreted as an abbreviation for "electronic mail".

Description

Both start-tag and end-tag are required. The content of <email/> is in one of two forms. The first consists of two required immediate subsidiary cements, <ad/> and <net/>, and optional <cmt/> elements and is used for representing addresses on most systems. The second is specific to the address formats of the international standard "MOTIS" or "X.400" messaging systems.

Format 1

The <email/> element includes two required immediate subsidiaries, the <:ad/> and the <net/>, and an optional <cmt/>, in that order. There is no immediate data. Together, the element and its subelements specify how to reach an individual by electronic mail.

<Cmt/> is permitted both immediately subsidiary to <email/> and immediately subsidiary to <net/>. In the former context, it provides information about the user's use of the mailbox or special addressing provisions. In the latter, it provides information about how the network itself is accessed.

Format 2

The <email/> element has one required immediate subsidiary element, <x400/>, and optional <cmt/> elements, in that order. There is no immediate data. The <cmt/> element is used, as in the first format, to provide information about access to the relevant network or mail system.

Notes on Networks and Addresses

The world is gradually developing two electronic mail addressing systems, with increasingly transparent gateways between the various networks that participate in each system and, of course, gateways between the two. One of these is the "domain name system" used in the National Research Internet environment in the United States and the systems attached to or imitating it (in a mail context, these are often referred to, incorrectly, as "RFC 822 addresses"). The other is associated with the international interconnection of systems using various profiles of the CCITT "X.400" or ISO "MOTIS" protocols.

The <ad/> and <net/> elements are associated with the first of these forms. They are optimized for a style of addressing often described as "a user on a host". Prior to X.400, this was essentially the only model in use, with variations on different networks. X.400 uses a structure of named (actually tagged) identifiers, and does not match the older model well.

Gateways and similar interconnections now exist between most of the networks listed under <net/>. If known and feasible, addresses should be listed as on hosts with registered Internet

Domain Names, and the "network" identified as "Inet", rather than distinguishing among the various specific networks. For example,

<email/> <ad/> jck@mitvma.mit.edu</ad/> <net/> INET </net/> </email/>

is preferred to

<email/> <ad/> jck@mitvma </ad/> <net/> BITN </net/> </email/>

although the two are, in most practice, identical. Similarly,

<email/> <ad/> infoods@mcimail.com </ad/> <net/> INET </net/> </email/>

is preferred to

<email/> <ad/> infoods </ad/> <net/> MCIML </net/> </email/>

Hosts that use the UUCP protocol and that are part of the mapping project (and no others) should use domain names (and <net/> Inet </net/>) if those names are registered, and the "host. UUCP" form with <net/> UUCP </net/> otherwise. UUCP hosts that are not part of the mapping project must provide "bang paths" from well known hosts.

Finally, X.400 electronic mailboxes that can be reached from the Internet or associated systems (including BITNET, EARN, etc.) should be specified in terms of Internet addresses, although X.400 may also be supplied if that is convenient.

Examples

<email/>
<ad/> Joe Smith <JSMITH@INFOODS.SOMEINSTITUTION.SOMENET>
</ad/>
<net/> Inet </net/>
</email/>

<email/>
<ad/> 76244,305 </ad/>
<net/> CompuS </net/>
<cmt/> telephone or telex after using; this address is rarely checked </cmt/>
</email/>

<email/> <ad/> infoods@infoods.mit.edu </ad/> <net/> Inet </net/>
<cmt/> Preferred electronic address </cmt/> </email/>
<email/>
<ad/> infoods@mcimail.com </ad/> <net/> Inet </net/>
<cmt/> Accesses different address from preferred one </cmt/> </email/>

The last example shown illustrates the use of multiple <email/> elements to include multiple electronic mail addresses for the same organization.

<ad/>

The <ad/> element is a required immediate subsidiary element of the <email/> element when the first format is used. It specifies the sender's electronic mail address.

Description

Both start-tag and end-tag are required. The content of <ad/> is a single unformatted data item: any character string not including </ad/>.

Format

The <ad/> element consists of a single string of characters comprising the address to which electronic mail for the sender may be sent.

Unlike ordinary unformatted data, the data in the <ad/> content can generally include "<" and ">"; only the contiguous string of characters </ad/> (which is not likely to be an exact substring of anyone's electronic mail address) is excluded from the content-it would be recognized as the terminating end-tag. As a result, <ad/> cannot include <cmt/> elements; they would be taken to be part of the electronic mail address itself. Therefore, if a <cmt/> element providing information about the electronic mail address is needed, it is made subsidiary to the containing <email/> element (see the examples under <email/>, above).

Example

```
<email/>
<ad/> Joe Smith <JSMITH@INFOODS.Someinstitution.Somenet> </ad/>
<net/> INET </net/>
</email/>
```

<net/>

The <net/> element is a required immediate subsidiary element of the <email/> element when the first format is used. It specifies the network for which the associated electronic mail address is intended.

Description

Both start-tag and end-tag are required. The content of <net/> includes a single required formatted data item, a keyword from the following extensible list (in either upper or lower case, or a combination

NETWORK	KEYWORD	NETWORK	KEYWORD
Internet	INET	JANET	JANET**
BITNET, EARN, etc.	BITNET*	MCIMail	MCIMail*
UUCP	UUCP*	OnTyme	ONTYME
SPAN	SPAN*	BIX	BIX
psi (DECNet)	DECPSI	CompuServe	COMPUS*
Sprintmail	SPRINT*	Fido	FIDO*

*At the time of this writing, good gateways to the Internet exist, and many hosts have domain name system addresses. These should be used if possible; see the discussion above under <email/>.

** Please reverse the address (i.e., change UK.AC.XXX to XXX.AC.UK) and designate as <net/> INET </net/> if the appropriate gateway connections are operable.

<Net/> may also include optional <cmt/> elements. If <cmt/> is used, the comment refers to the network itself, not to the overall electronic mail address and how it is used. Compare the last two examples below.

In using an electronic mail address, the important issue is addressing from somewhere else, and, in particular, somewhere from which the receiver of a file can reach the addressee. Many of the "networks" listed above are not really networks but single systems that people log into, however remotely, to send and receive mail. If you list the name of a disconnected network, please indicate, with a <cmt/> element, how it can be accessed. See the discussion of "Networks and Addressing" under <email/>, above, for more information.

Format

The <net/> element consists of a single string of characters which is an address to which electronic mail for the sender may be addressed.

Examples

<email/>
<ad/>
Joe Smith <JSMITH@INFOODS.SOMEINSTITUTION.SOMENET>
</ad/> <net/> INET </net/>
</email/>

<email/> <ad/> Doe@somehost.span.nasa.gov </ad/> <net/> INET </net/>
</email/>

<email/> <ad/> Somehost::Doe </ad/> <net/> SPAN </net/> <cmt/> SPAN form of Internet address </cmt/> </email/>

<email/> <ad/> foodtable@agri.govt.fj </ad/> <net/> Fijinet
<cmt/> At present, not accessible from outside Fiji </cmt/> c /net/> </email/>

Of the last two examples, the first illustrates a comment that is applicable to the electronic mail address, specifying its relationship to other supplied addresses. The second one applies to the network, and specifies accessing information or the lack thereof.

<x400/>

The <x400/> element is a required immediate subsidiary element of the <email/> element when the second format is used. It specifies the sender's electronic mail address. If the address is accessible from the Internet, <email/> should be used twice, once with the <ad/> and <net/> elements to specify the address path from the Internet, and once with <x400/> to specify the actual X.400 address.

Description

Both start-tag and end-tag are required. The content of <x400/> is a single unformatted data item: any character string not including </x400/>. Information equivalent to the <net/> element associated with <ad/> is, of course, supplied by the country and primary management domain fields.

Format

The <x400/> element consists of a single string of characters comprising an address to which electronic mail for the sender may be sent.

Unlike ordinary unformatted data, the data in the <x400/> content can generally include "<" and ">"; only the contiguous string of characters </x400/> (which is not likely to be an exact substring of anyone's electronic mail address) is excluded from the content-it would be recognized as the terminating end-tag. As a result, <x400/> cannot include <cmt/> elements; they would be taken to be part of the electronic mail address itself. However, a <cmt/> element may be used as a subsidiary to the containing <email/> element, so this should not be a major restriction.

At the time of this writing, the form in which an X.400 address should be written for people to read (the "presentation format") has not been standardized and differs from one system to another. Until there is a Standard, any reasonable format that identifies the pairs of keywords (tags) and values may therefore be used; the one shown in the example below is preferred.

Example

```
<email/>
<x400/>
OU = Rocquencourt;O = INRIA;P =ARISTOTLE;A =ATLAS;C = FR
</x400/> <cmt/> Internet address given reaches the same mailbox </cmt>
</email/>
```

<phone/>

The <phone/> element is an optional, repeatable immediate subsidiary element of the <sender> and <source> elements. It specifies the sender's or source's complete telephone number, in international form. A comment may be added to document local conventions, times of day, etc.

Description

Both start-tag and end-tag are required. The content of <phone/> consists of one unformatted data item and optional <cmt/> elements.

Format

The <phone/> element's unformatted data consists of a single string of characters that constitute the sender's or source's telephone number. The entire international telephone number should be given; this includes country and other appropriate prefixes. If an extension number is needed, it should be given after an "X".

While the form is still not in general use in some countries, the international convention for writing a phone number consists of a plus sign (+) to denote any local access code for international service, the country code, a space, the city code (if any), a space, and the local number. Spaces may used in the local number according to local conventions for separating parts of numbers, but no additional symbols (such as dashes or parentheses) are used.

Examples

```
<phone/> +1 617 253 8004
<cmt/> INFOODS Secretariat main number in the USA. </cmt/>
</phone/>

<phone/> +1 1013214567 X 123 </phone/>

<phone/> + 64 63 68019
<cmt/> Use (063) within country but outside city </cmt/>
</phone/>

<phone/> +255 51 28951
<cmt/> May be answered either by a fax machine or a person </cmt/>
</phone/>
```

<fax/>

The <fax/> element is an optional, repeatable immediate subsidiary element of the <sender> element. It specifies the sender's international telephone number for receiving facsimile transmissions. See the discussion of <phone/> for additional information about the expression of telephone numbers.

Description

Both start-tag and end-tag are required. The content of <fax/> consists of one unformatted data item and optional <cmt/> elements.

Format

The <fax/> element's unformatted data consists of a single string of characters that constitute the sender's fax telephone number. The entire international telephone number should be given; this includes country and other appropriate prefixes.

Examples

```
<fax/> +1617 253 8000
<cmt/> For voice confirmation of transmission, call + 1 617 253 3690. </cmt/>
</fax/>

<fax/> +99 1 22 33 55
<cmt/> Person will answer; you must ask for fax machine. </cmt/>
</fax/>
```

<telex/>

The <telex/> element is an optional, repeatable immediate subsidiary element of the <sender> and <source> elements. It specifies the sender's telex number.

Description

Both start-tag and end-tag are required. The content of <telex/> consists of one unformatted data item, an optional <ansbk>, an optional <sys> and optional <cmt/> elements. The latter may be used to specify special information about the use of the telex address. <Ansbk> and <sys> are defined below.

Format

The <telex/> element's unformatted data normally consists of a single string of characters that constitute the sender's telex number. The answerback, if available, should be specified with the <ansbk> tag and follow the telex number itself. The <sys> subsidiary element specifies which telex system is to be used, if more than one is available in the country concerned. It will not be used for countries that have a single telex system.

Local Subelements

<Ansbk> is optional and is used to specify an answerback string. The content consists of an unformatted string and terminates when another tag is encountered.

<Sys> is optional and is used to specify the applicable telex system. The content consists of an unformatted string and terminates when another tag is encountered.

Example

<telex/> 6502688345
<ansbk> 6502688345 MCI UW
<sys> WUI/MCI
<cmt/> INFOODS main telex number. Warning: goes through computer, checked only weekly </cmt/>
</telex/>

<cable/>

The <cable/> element is an optional, repeatable immediate subsidiary element of the <sender> and <source> elements. It specifies the sender's international cable address.

Description

Both start-tag and end-tag are required. The content of <cable/> consists of one unformatted data item and optional <cmt/> elements.

Format

The <cable/> element's unformatted data consists of a single string of characters that constitute the sender's international cable address.

Examples

```
<cable/> MITCAM </cable/>
<cable/> INMU BANGKOK
<cmt/> Temporary, 881010-890701 </cmt/>
</cable/>
```

The second example uses a <cmt/> element to specify something about the cable address. Since the comment associated with this element is free text, the dates specified are not required to be in any particular format. Nonetheless, the format given for dates is preferred to the one shown here, since it is internationally unambiguous and avoids end-of-century problems.

<cmt/>

The <cmt/> element is an optional, repeatable immediate subsidiary element of various other elements. It should include only peripheral information related to the element to which it is subsidiary, and should not include any information for which a "real" interchange element exists or might reasonably be defined. "Cmt" can be thought of as an abbreviation for "comment".

Description

Both start-tag and end-tag are required. The content of <cmt/> consists of one special unformatted data item: any character string not including </cmt/> (see below). In general, <cmt/> elements are permitted subsidiary to any element that requires an end-tag and that accepts immediate data, and not otherwise.

Format

The <cmt/> element consists of a single string of characters. It is not interpreted by conversion programs or programs processing interchange files but is intended to supply ancillary information for the human user.

Unlike ordinary unformatted data, comment data can generally include " <" and "> "; only the contiguous string of characters </cmt/> is excluded from the content-it would be recognized as the terminating end-tag.

Examples

<cmt/> Temporary, 881010-890701 </cmt/>

<cmt/> Certainly <0.001 </cmt/>
<cmt/> The value shown with <energc> is probably more representative </cmt/>

The first example above shows the use of a "less-than" sign in a comment. In general, this symbol is not permitted in free text; <cmt/> elements are among the exceptions. The second shows the use of something that appears to be a tag but is not: its presence as part of a <cmt/> element causes it to be treated as ordinary text. While this type of construction is permitted, it is discouraged because it will almost certainly confuse users. It would be better to write, e.g.,

<cmt/> The value shown with "energc" is probably more representative </cmt/>

<source>

The <source> element is a structural element, the second immediate subsidiary element of the <header> element. It includes all of the information about the data base from which the data in the interchange file was obtained, what restrictions apply to the use or publication of the data, and who is responsible for the data.

Description

Both start-tag and end-tag are required. The content includes one required immediate subsidiary element (numbers in parentheses refer to the pages where the elements are described):

<ref/> (55)

and these optional immediately subsidiary elements:

<fullname> (33)	<postcode> (40)	<telex/> (50)
<ianame/> (35)	<title/> (41)	<cable/> (51)
<orgz> (36)	<email/> (42)	<cmt/> (52)
<contact/> (37)	<phone/> (48)	<restrict/> (59)
<addr/> (38)	<fax/> (49)	<sourceref> (60)
<country> (39)		

In some cases, a data base may exist in both table and computer-processable form, and it may be desirable to cite both. When this occurs, the special delimiter "<->" may be used to separate the two (or more) citations within a single <source>. When two or more references are given this way, they do not imply separate sources, e.g., a combination of tables, but different forms of availability of the same conceptual data.

The <source> element's immediate subsidiaries are not structural elements, so they may appear in any order. There is no immediate data.

Format

The <source> element has subsidiary elements giving information about the source of the data in the interchange file.

All of the data in an interchange file must be from a single source; if data from two or more sources are to be sent to the same receiver, they must be separate interchange files (i.e., separate <infoods 85> elements; these may be combined into a single physical file or message for transmission). This restriction is closely connected with the concept of "authority" as discussed in Chapter 1, and is not as limiting as might at first appear. For example, assume that the food tables for country A were made up from local (i.e., country A) data plus some data incorporated from the data bases of countries B and C. If these data were re-exported in interchange form, the source *for the data base* would be the tables of country A, reflecting the authoritative decision to combine the three sets of values, decisions as to which foods were drawn from each table, etc. However, if a sender in country A chose to re-export the data from countries B and C intact and unedited, the restriction stated here would be relevant, and two interchange files (one for each source) would be required.

The data source is identified in the <ref/> element. The data may come with some restrictions on its use or publication; these are specified in the <restrict/> element. The rest of the immediate subsidiary elements serve to identify the person or corporate entity who, by virtue of compiling the laboratory data, deriving, extracting, or collecting the data from other sources, or publishing the data, is effectively responsible for their content and substance. The intent of these elements is similar to their use subsidiary to <sender/> and they may duplicate data in the <ref/>, especially if the data source is a publication.

The origins and derivations of values for individual foods are reflected in the international food record identifier and related information discussed in Chapter 5.

Example

<source>
<ref/> *information identifying the source data base* </ref/>
<fullname> *whoever Is responsible for the data* </fullname>
additional elements providing information about the person or organization responsible
</source>

In this example the data are provided with no restrictions on their use or publication. If restrictions did exist, the <restrict/> element would be included.

<ref/>

The <ref/> element is a required immediate subsidiary element of the <source> element. It identifies the source data base.

Description

Both start-tag and end-tag are required. The <ref/> content includes an unformatted data item and the following optional immediate subsidiary elements (numbers in parentheses refer to the pages where these elements are defined):

<cmt/> (52)	<isbn> (57)
<pub/> (56)	<issn> (58)

Format

It has traditionally been difficult to determine how to reference a data base in conventional publications (e.g., journal articles and books). This element specifies the form of reference preferred by the source of the data. If the data come from, or have appeared in, published form, the <pub/> element gives more information about the publisher, and <isbn> or <issn> gives standardized identification information about the publication.

Machine-readable data bases may be referenced as well as printed ones, and that usage should increase over time. However, to be useful, they must be "published" in the sense of being available from a source, at least for reference, and the information supplied with this element must be sufficient to identify the data base and locate a source of availability. The list of elements permitted to appear subsidiary to the <ref/> element is likely to be expanded as mechanisms for referencing, archiving, and distributing machine-readable data bases (other than as variations on printed publications) become more established.

Examples

<ref/> US Department of Agriculture. Composition of Foods: Finfish and Shellfish Products. Agriculture Handbook 8-15.</ref/>

<ref/> Paul AA and Southgate DAT. McCance and Widdowson's The Composition of Foods, 4th edition. <pub/> London: Her Majesty's Stationery Office </pub/> <isbn> 0 11 450036 3 <ref/>

<pub/>

The <pub/> element is an immediate subsidiary element of the <ref/> element. It identifies the publisher if the source data base is published.

Description

Both start-tag and end-tag are required. In addition, successive lines of <pub/> are separated by the special tag <->. Each line consists of one unformatted data item.

Format

<Pub/> is an element whose content is successive lines of a publisher's name and address, separated by the special tag <->.

Examples

```
<pub/> Addison-Wesley
<-> Reading, Massachusetts
<-> USA
</pub/>

<pub/> Technique et Documentation - Lavoisier <-> 11, rue Lavoisier
<-> F75384 Paris Cedex 08 <-> France
</pub/>
```

<isbn>

The <isbn> element is an immediate subsidiary element of the <ref/> element. It identifies the ISBN of the referenced book. "ISBN" is the standard abbreviation for "international standard book number".

Description

The start-tag is required; there is no corresponding end-tag. The content of <isbn> consists of one unformatted data item that ends when another tag is encountered.

Format

The <isbn> element's unformatted data consist of a single string of characters that constitute the referenced publication's ISBN as established by ISO 2108.

Example

<isbn> 0-201-134489

<issn>

The <issn> element is an immediate subsidiary element of the <ref/> element. It identifies the ISSN of the referenced serial or journal. "ISSN" is the standard abbreviation for "international standard serial number".

Description

The start-tag is required; there is no corresponding end-tag. The content of <issn> consists of one unformatted data item that ends when another tag is encountered.

Format

The <issn> element's unformatted data consists of a single string of characters that constitute the referenced publication's ISSN number as established by ISO 3297.

Examples

<issn> 8750-6874
<issn> 0889-1575

<restrict/>

The <restrict/> element is an optional immediate subsidiary element of the <source> element. It is used to list any restrictions on the distribution or use of the data base or other distributed material.

Description

Both start-tag and end-tag are required. The <restrict/> content consists of unformatted text.

Format

The content of the <restrict/> tag consists of unformatted text describing the restrictions that apply. A <cmt/> element may be used if needed.

Example

<restrict/> Data protected by copyright, royalties required. This file may be studied, and values used for imputation of data for other tables as long as the source is acknowledged. Other uses require making arrangements with the data source.
</restrict/>

<sourceref>

The <sourceref> element is an optional, but very desirable, immediate subsidiary of the <source>. It specifies the way this interchange file should be referred to when communicating with the source individual or organization.

Description

The start-tag is required; there is no corresponding end-tag. The content of <sourceref> consists of one unformatted data item that ends when another tag is encountered.

Format

The content of <sourceref> is unformatted data which must consist of characters making up a name or phrase by which this collection of data can be identified in communications to the source person or organization; it is especially useful in identifying special-purpose subsets of the data base described in <ref/> . It is optionally provided by the source, or may be provided by a regional centre on behalf of the source.

Example

<sourceref> Stanford special extras of USDA Handbook 8, Jan 1988

<dflt>

The <dflt> element is an immediate subsidiary of the <infoods 85> element. It is optional, but, if it appears, it does so as the second subsidiary element (after <header>). It prescribes defaults for various elements or parts of their content. These defaults are applied throughout the interchange file.

Description

This is a structural tag, and both start-tag and end-tag are required. The structure of the content is precisely the same as that for the <food> element, but the asterisk character may be used to indicate data value positions.

Format

The content of <dflt> has precisely the same structure as a <food> element, except that all subsidiary elements are optional. <Dflt> itself must be omitted if all immediately subsidiary elements are omitted. If <dflt> and a subsidiary element are used to specify a data value for a particular position, no other value may appear in that position. Details of the relationship between elements subsidiary to <dflt> and those subsidiary to <food> and of how <dflt> is applied are discussed in Chapter 3.

While default values will most often be used subsidiary to <comp> or <drvd-comp> to specify, e.g., units of measure for particular nutrients, default values for subsidiaries of <classif> or even <fddflt> are plausible in some cases, especially when very specialized tables or homogeneous laboratory data are involved. For example, if an interchange file consisted entirely of data about samples of "Granny Smith apples" and all measurements were to be reported both per 100 grams edible portion and per apple, the second example shown below illustrates a possible interchange file structure.

Examples

```
<dflt> <comp> <procnt> * JONES </procnt> <energc> * KJA </energc> </comp> </dflt>
```

```
<dflt>
<fddflt> * <-> <meas/> Per fruit </meas/> </fddflt>
<classif> <bvname> Granny Smith Apple </bvname> </classif>
</dflt>
```

5. The food element and subelements

INTRODUCTION

This chapter contains descriptions of the elements that describe a food and its components, using the same format as chapter 4. The elements that identify the food components themselves appear in a separate publication [17], and elements that describe the data (e.g., the statistics being reported and the quantities of food measured) appear in chapter 6.

<food>

The <food> element is an immediate subsidiary of the <infoods 85> element. If <dflt> is used, the first <food> element will come third in the sequence of those immediately subsidiary elements. Each <food> element includes data about a single food. There may be any number of <food> elements in an interchange file.

Description

This is a structural tag, and both start-tag and end-tag are required. The content is a <classif> element, optionally followed by <fddflt>, <comp>, and/or <drvd-comp> in that order; this list is extensible. Normally at least one of <comp> or <drvd-comp> will be present, but for special purposes a valid interchange file might contain food description (classification) information only. All immediate subsidiaries of the <food> element are structural elements.

Format

The <food> element contains all the data in the interchange file about a single food. All of the identifying information is in the <classif> subsidiary element; the data about directly measured and derived components of the food are found in the <comp> and <drvd-comp> elements. If there are any defaults to be applied on a per-food basis, they are specified in the <fddflt> element.

The distinction between <comp> and <drvd-comp> information is whether or not the food "component" in question is a real, directly measurable component, or is instead a measure that is computed from other "real" components. This decision is made when the generic identifier for the component is established. See Identification of Food Components for INFOODS Data Interchange [17] for additional discussion.

Each component has a tag/generic identifier separately established (by registration) that identifies which component is being reported on. But each food does not: new foods can be accommodated into interchange files at will, whereas new components of interest must be registered. The result of this is that there must be associated with each <food> a collection of identifying data. This is the reason for a separate <classif> component with a large number of possible subelements.

Example

```
<food>
<classif> <ifri> International Food Record Identifier </ifri>
<bvname> identifying name of the food </bvname>
other identifying information
</classif>
<comp> various food component data elements </comp>
</food>
```

<classif>

The <classif> element is the first immediate subsidiary of the <food> element. It contains information that distinguishes the particular food to which the component data within the <food> element refers from other foods.

Description

Both start-tag and end-tag are required. The content includes one required immediate subsidiary element (numbers in parentheses indicate the page numbers on which the element descriptions begin):

<ifri> (66)

and one or more of these optional subsidiary elements:

<srcdbid> (69)	<exname> (72)
<bvname> (71)	<image> 75

or any of the set of elements identified in this document or by subsequent registration as <*specific classification*> (77) or <*food description*> (78). (At least one <bvname> element should be included, possibly associated with an <exname>.) Each of the optional elements may be repeated; <ifri> may appear only once.

<Classif> is a structural tag, so all elements subsidiary to it require end-tags and do not have trailing slashes in their generic identifiers.

Format

The <classif> element identifies the particular food to which the component data within the <food> element refers, and the source record of that data. In particular, the International Food Record Identifier uniquely identifies the food and the data source for the component data included for that food in the interchange file. The IFRI should always be given (see <ifri> in this section). The <srcdbid> elements provide any record identifiers other than the IFRI that may be appropriate. <bvname> and <exname> elements provide various names by which the food is known.

All elements immediately subsidiary to <classif> are treated as structural i.e., they do not contain a trailing slash in their generic identifiers but nonetheless require end-tags.

Example

```
<classif>
<ifri> UN.FAO.NEast82.III.58 </ifri>
<bvname> Milk, Cow, fluid, whole </bvname>
</classif>
```

<ifri>

The <ifri> element is an immediate subsidiary of the <classif> element and the only required subsidiary element of <classif>. It specifies the International Food Record Identifier of the data about the particular food being reported. As discussed below, the International Food Record Identifier, sometimes called the "IFRI" or, less formally, the "food record identifier" or "record identifier", is a critical concept for the monitoring of data as they pass between data bases and tables.

Description

Both the start-tag and end-tag are required.

Format

The content of <ifri> is a single unformatted data item consisting of that character string which is the registered International Food Record Identifier of the data included in the <food> element of which <ifri> is a subsidiary.

Assigning the International Food Record Identifier

While the interchange system can be used for other purposes such as data storage and management or interchange within regions, when interchange files are passed between regions, this element must be supplied and must be valid. The criteria for validity are simple: the food record identifier must be internationally unique, and must be able to be used to reconstruct the data source.

In order to assure this, the record identifier itself is formatted, using ordered facets, the restricted ISO 646 character set, and no embedded blanks or other whitespace characters. The first of those facets is a two- or three-letter code for the region or international organization; these codes are assigned by the INFOODS Secretariat, in consultation with the regional data centre or organization involved. The structure of any remaining facets is the responsibility of the region, as long as uniqueness is guaranteed. We expect that most regions will use a system that results in record identifiers with a structure similar to:

Region. Country.Publisher. SpecificDataBaseAndDate.SequenceNumber with "publisher" replaced by "laboratory" or "agency" as appropriate. That structure is used in the examples in this book, but remains optional. This type of approach also makes it possible for a regional centre to further delegate the assignment of food record identifiers to individual data compiler organizations by assigning each organization a leading set of facets and then permitting them to assign their own sequence numbers or, possibly, additional intervening facets.

Where it is sensible, regions are urged to adopt the conventions and identifiers of the ISO system for the identification of organizations [46]. That system uses a similar structure, with more global registration authorities (e.g., national bodies) assigning their own identification and then delegating assignment of particular identifiers (e.g., to different state or provincial organizations).

When dealing with older data, it may be very difficult to determine whether a food record is "original" or whether it is a CODY from another data base. When no determination can be made, a new food record identifier should be assigned; it is better to make errors in the direction of asserting that two records arc different when they arc not than to assert identity when it does not exist. From the user's point of view, this implies that identical food record identifiers-an assertion that data values in records are the same-can be trusted as evidence that the data are identical and come from an identical source. Differing record identifiers are to be taken as no more than a strong hypothesis that the data have different origins. Whether different food record identifiers are to be trusted more or less as an indication that the corresponding data records represent different origins, e.g., separate analyses, must be evaluated on the basis of other information.

Nothing in the system implies that a region cannot assign a food record identifier of the form:

Region.wdkwtcf.Sequence where "wdkwtcf" can be read as "we don't know where this came from". Again, this is acceptable as long as the record identifier is unique and the regional centre will take responsibility for it. Once the record identifier is assigned, everyone else knows where the data came from: whoever in the region assigned the record identifier.

Identical food record identifiers in two different interchange files do not, however, imply that the two interchange file <food> elements are identical. If a data base compiler inserts food 75 in table B. which contains only proximate values, by copying the proximates (only) from food 200 in table A, then the food record identifier provided when table B is passed between regions should reflect food 200 in table A. This is true even if that identifier in table A refers to a record-a <food> element-that contains many other nutrients.

For composite records, for example, the compiler might use proximate data from food 201 in table A, but substitute protein data from food 3 in table C, to build food 76 in table B. The food record identifier in table B should be a new one, associated with table B. The combination of values from the two other tables causes the food record in table B to be "original values" for which the developers of table B must take responsibility. The principle used here is discussed in more detail in Chapter 1.

If an interchange file is used for purposes other than data interchange between regional centres, we recommend that valid food record identifiers be assigned nonetheless. Errors are much less likely if the record identifiers are assigned as early and as close to the time and point of first "publication" (circulation of data values outside the laboratory or organization where they are compiled) as possible. If, for some reason, it is desired to retain or transport data in interchange form without the food record identifiers attached, use of <srcdbid> is recommended to transmit any other food record identifying information.

Finally, it is important to understand that the food record identifier does not, in any usual sense, identify a "food"; it identifies a particular collection of data, a "food record". In the most extreme case, an interchange file might contain nothing but a collection of raw laboratory data on samples of a single food. One might then use the <dflt> element to specify name and classification information for the entire interchange file, since this information would be identical.

On the other hand, since the individual <food> elements would contain different data, their food record identifiers would be different. If those data were ultimately aggregated to yield a single set of values for the food in another table, that set would be assigned a new food record identifier, since the process of aggregating the raw laboratory data yields a new set of values.

The International Food Record Identifier value is intended to serve two purposes: for the nutritionist, the value should evolve to provide a definitive answer to the question

"Is this value copied from another table and, if so, which one?"; for the data base manager, it is a key that guarantees uniqueness of each food record that is, in fact, unique.

Assignment of Regional Identifiers

Actual identifiers for regional groups and other delegated IFRI allocators will be assigned by the INFOODS Secretariat. Identifiers assigned at the time of this writing are shown in Appendix A.

Examples

Since EUROFOODS has not, at the time of this writing, created an IFRI system, the first example is only illustrative of what might be done.

<ifri> ER.UK.M-W78.171 </ifri>
<ifri> UN.FAO.EAsia72.170 </ifri>

The second example designates the data record associated with boiled, sweetened adzuki beans in the 1972 FAO Food Composition Table for Use in East Asia [9].

<srcdbid>

Each <srcdbid> element is an optional immediate subsidiary of a <classif> It identifies a food and its data with respect to the source data base. "Srcdbid" may be thought of as an abbreviation for "source data base identifier". However, it applies at and is subsidiary to the <food> level, and consequently identifies a food record in the source data base, not the source data base as a whole.

Description

Both start-tag and end-tag are required. The content of <srcdbid> consists of one unformatted data item, followed by optional immediately subsidiary <cmt/> elements. This element is optional, and is provided for the convenience of data base compilers in keeping track of the relationship of their data bases with interchange files prepared from them. The actual content of a specific <srcdbid> is at the discretion of the data base compiler or organization preparing the interchange file.

Format

Each <srcdbid> relates a record in the original data base to the data contained within the <food> element to which this element is subsidiary. In most tables and data bases, the final identification of food information is a sequence number, possibly used in combination with a page number or its equivalent. That sequence number may explicitly reflect a food grouping, as in the USDA Standard Reference Database [35], or may just reflect the sequential organization of various versions of the table, as in recent editions of McCance and Widdowson [23].

In most cases, this sequence number will be encoded in the required international food record identifier (see the <ifri> description). However, it need not be: as discussed under that element, assignment of those identifiers is left to regional decisions and conventions. If the sequence number is not incorporated in the international food record identifier and it is desired to keep track of it, or if it is desired to isolate it from the international food record identifier for some other reason, this element would typically be supplied, with the sequence number as its content.

The example below illustrates the relationship in the more typical case, where the sequence number is also coded into the international food record identifier. There are at least two other situations, in which this correspondence is less likely, and a separate <srcdbid> would be important:

- If a volume of data were organized into tables so that foods were listed and numbered within the individual tables (as in the FAO tables for the Near East [10]) and the IFRI were assigned by numbering foods throughout the tables rather than on a per-table basis, <srcdbid> might be used to show the table-numbered food relationship.
- If a volume of data, or a file to be interchanged, consisted largely of values borrowed from many other tables, <srcdbid> might be used to show the food numbers in the compendium, while the international food record identifier showed the sources of the data records.

Example

```
<ifri> UN.FAOAfrica68.55 </ifri>
<srcdbid> 55 </srcdbid>
```

<bvname>

The <bvname> element is a repeatable immediate subsidiary of the <classif> element. It specifies a name by which the associated food is known, all of whose characters are drawn from the character table specified in ISO 646 as the "Basic Version" [40]. It is complementary to <exname>, which permits a broader selection of characters. "Bvname" can be thought of as an abbreviation of "basic-version name".

Description

Both start-tag and end-tag are required. The content is an unformatted data item, optionally followed by a <lang> element and by optional <cmt/> elements, which would typically be used to describe the use of the particular name.

Format

The <bvname> element contains a single unformatted data item consisting of the characters that make up a name for the food whose data is within the <food> element to which this element is subsidiary. This name must be in or transliterated into the restricted ISO 646 character set normally required of all data in an interchange file. <Lang> may be used, if desired, to designate the language in which the name appears (directly or in transliteration). <Bvname> differs from <exname> in that the latter can support names written in any character set.

Example

```
<bvname> cake <lang> en
<cmt/> May not be an exact translation, the French is correct </cmt/>
</bvname>
<exname> gâteau <lang> fr <charset> 8859 1 </exname>
<exname> wienerbrød <lang> da <charset> 8859 1 </exname>
```

This example illustrates the use of <bvname> and <exname> together to express the name of a food in both the minimal character set and a character set more suited to another language. <Bvname> should be used for "cake", since that uses only basic characters; <exname> is required for the other names.

If all three of the names above were used as part of the <classif> element for the same food, they might illustrate the difficulties in "translating" this type of name.

<exname>

The <exname> element is a repeatable immediate subsidiary of the <classif> element. It specifies a name by which the associated food is known that uses an extended (rather than the basic) character set. "Exname" can be thought of as an abbreviation for "extended name".

Description

Both start-tag and end-tag are required. The content is an unformatted data item, followed by both <lang> and <charset> immediate subsidiaries, which are required. The content may also include one or more <cmt/> elements, typically used to specify the context in which the name is used. <Exname> differs from <bvname> in that the latter uses a restricted Latin-based character set, while <exname> permits a wide range of character sets and languages.

Format

The <exname> element consists of a single unformatted data item consisting of the characters that make up a name for the food whose data is within the <food> element to which this element is subsidiary. It is used when this name requires a character set other than the ISO 646 Basic Character Set [40] to which most of the elements of the interchange file are restricted, as discussed in Chapter 3. The name may be expressed in any language recognized by ISO for which an appropriate computer character coding exists. That data item is followed by required <lang> and <charset> subsidiary elements and may be followed by one or more <cmt/> elements.

The language must be specified by the <lang> immediate subsidiary element. The character set must be specified by the <charset> subsidiary element.

Except in special circumstances, <exname/> should not appear without <bvname> at the same level; since many receivers of interchange files will be unable to completely process names in international alphabets, names should always be provided transliterated into the standard restricted alphabet as well as in their original alphabet.

See the examples under <bvname> on the previous page and under <lang> and <charset> on the following pages.

<lang>

The <lang> element is an immediate subsidiary of any element whose data item may be in a specifiable language. In particular, it is an optional subsidiary element for <bvname> and a required subsidiary element for <exname>.

Description

The start-tag is required; there is no corresponding end-tag. The content of <lang> consists of either one keyword or an unformatted string and ends when another tag is encountered.

Format

The content of c <lang> is a keyword drawn from the ISO 639 [39] lower-case two-letter code for the language intended. If no such code exists, then the language may be described by a name or phrase of more than two characters, which must be expressed in the restricted ISO 646 character set generally permitted for interchange file data. If ISO 639 contains a code for the language, the code, rather than a phrase, should be used.

Example

<exname> nød <lang> da <charset> 88591 </exname>

See the description of <bvname> for additional examples.

<charset>

The <charset> element is an immediate subsidiary of those specific elements whose data are permitted to be in character sets other than the ISO 646 character set. <Exname> and <ianame/> are the only such elements in the initial edition of this list. <Charset> may not be used without <lang>

Description

A start-tag is required; there is no corresponding end-tag. The content of a <charset> element consists of two formatted data items, both numerals.

Format

Any element permitting <charset> as an immediate subsidiary must have an unformatted data item; the <charset> element specifies a character set (other than the restricted subset of ISO 646 that is normally required) by which the unformatted data item is to be interpreted.

At present the only alternative character sets permitted are those standardized in the ISO 8859 series [48-52]. Until and unless the definition of this element is expanded, the first numeral of the <charset> content must be "8859". The second data numeral designates which of the registered character sets is intended (they are numbered by the ISO standardization process).

Currently, all ISO 8859 character sets retain the "less-than" and "greater-than" signs in their ISO 646 positions. Only such character sets are acceptable under this interchange specification, in order to preserve the ability to easily recognize tags.

Use of character sets other than the ISO 8859 group is not currently anticipated, although a universal character-set standard may be permitted if one comes into general use. The "8859" data item is nevertheless required to permit extension to include other standards if it should become desirable in the future (such extension would require registration [see chapter 7] and amendment of the description of the element) and as a contingency against future changes in the way standard character sets are organized and registered.

Example

<exname> ТВОРОГ JIPHbI <lang> ru <charset> 8859 5 </exname>

See the description of <exname> for additional examples.

<image>

The <image> element is a repeatable immediate subsidiary of the <classif>. It introduces one or more subsidiary elements which, in turn, provide a picture or drawing of the food.

Description

Both start-tag and end-tag are required. The content consists of one or more elements that indicate the picture encoding type and provide the actual image. A <cmt/> element may also be included.

Format

The <image> element contains elements only. The first of these elements must be an image format designating element (see below). Additional elements may appear or even be required depending on which image format element is chosen, but a particular <image> element may contain only one image format designating element. A <cmt/> element may be used, and should be supplied when possible, to describe the image. Images should be used with caution in interchange among regions: while a picture is often worth a thousand words, it may require the equivalent of many thousand characters to store. The large files this implies may be burdensome for some data receivers.

Image Format Designating Elements

At present, there are several different ways to represent pictures and drawings for interchange purposes but none of them have emerged as a clear standard, supported in most computer systems from which image display would be desired. The two formats described below are well-documented and widely available. However, this element may be extended by adding additional (alternative) image format designating elements in the future. Please inquire with the INFOODS Secretariat before using any image formats in inter-regional interchange of food composition data.

<g3fax2/>

<g3fax2/> indicates a content that is the encoded format of a Group III fascimile, as specified in CCITT Recommendation T.4. Only the two-dimensional format is supported. The content is the encoded image itself further encoded into the standard "base64" format so that all information transmitted is in character format and protected from unexpected network transformations. Line break characters are ignored within the content. The end-tag is required; other than <cmt/>, no other element of <image> may accompany <g3fax2/>.

<gif/>

<gif/> indicates a content that is a color graphic image encoded in the CompuServe "GIF' format [2]. The content is the encoded image itself further encoded into the standard "base64" format as discussed above. The end-tag is required; other than <cmt/>, no other element of <image> may accompany <gif/>.

Example

<image> <gif/> *image in base64 coding of the CompuServe GIF format* </gif/> <cmt/> leaf detail of Conium maculatum </cmt/> </image>

<specific classification>

The <*specific classification*> elements are immediate subsidiaries of the <classif> element. They identify specific food classification, nomenclature, or formal description systems that have entered international usage or that reflect well-documented national or local systems. The content of these elements is the classification, as dictaded by the particular system. Additional element types will be registered as needed, as discussed below.

The term <specific classification> is shown in italics as a reminder that it is not an actual tag and never appears in an interchange file but, instead, is a placeholder for a series of individual elements.

Description

Both start-tag and end-tag are required for all of these elements. The content will vary from one to the next, and will be specified by individual element registrations. The content may not contain the character "<" and should not contain either "/" or ">".

Elements of This Type and Their Registration

The criteria for the registration of a classification system are as follows:

- The system is in use, or is soon to be in use, in more than one country or other administrative unit.
- The system has a clear and stable definition. The definition document must be readily obtainable and should either be on file with the INFOODS Secretariat (with permission to reproduce if other sources become unavailable) or be published in widely available literature (e.g., in a book or journal with broad circulation).

The requirement for a stable definition is important because automatic comparisons among categories require that the coding be precisely known: accidental comparisons between different versions of a system can lead to significant errors. Consequently, systems that are still evolving in ways that might change older definitions will be registered only with version identification.

The following systems meet the above criteria, and have been registered prior to the publication of this document:

<EUROCODE2> The "Eurocode 2" system, as described by Arab et al. [1]. The format of the element is <EUROCODE2> *code sequence* </EUROCODE2>.

For example, a cola drink would be
<EUROCODE2> 12.10 </EUROCODE2>

<FDA-FFV 8807> The version of Langual (formerly the Factored Food Vocabulary) released for study and comment in July 1988 and documented in reference 13. Other versions of Langual can be registered as needed, upon the submission of comprehensive definitions.

<food description>

<*Food description*> includes elements that are derived from the philosophy and content of the open-ended INFOODS system for describing foods [31], 32]. While the INFOODS system for identifying food components [17] includes the specific generic identifiers to be used, the description system does not. Consequently, the element formats appear below, but a knowledge of the description system and its documentation is needed to understand them. When and if the description system is expanded, the list below will be extended as well. All of these elements are immediate subsidiaries of the <classif> element.

Additional elements may be registered as needed, independent of general revisions to the description system. Since these elements will, in general, contain informal text, the requirements for registration will be a demonstration of need for additional ones and a description of the content.

The term <food description> is shown in italics as a reminder that it is not an actual tag and never appears in an interchange file but, instead, is a placeholder for a series of individual elements, which are listed below.

Description

Each of these elements requires both start-tag and end-tag. The content will, in general, consist of unformatted text, unless a different format is specified below or in the description system. <Cmt/> elements can be used as needed.

In particular, many of the elements require keywords from the open-ended keyword list associated with the description system. If additional terms are used, they should be supplemented with <cmt/> elements.

Generic Identifiers and Descriptive Notes

The "section" listed below refer to the documentation of the INFOODS description system [32, Table 2, pp. 25 ff]. That discussion contains information about the use, format, and content of each of the elements listed below. The information below constitutes supplemental material to assist in translating between the description system documentation and the corresponding interchange system elements.

Section A: Source of food, names, descriptive terms.

<fdsource/> Source and sampling information specific to the food. Information related to the data base, laboratory, etc. should be supplied as elements subsidiary to <header>. In particular, the elements subsidiary to <source> may appear here if needed.

<fdagg/> Description of how aggregate foods were aggregated. The content is unformatted text.

Section B.: Name and identification of food

This section is used to specify the information that is usually thought of as "names of foods". These names are specified subsidiary to <classif> using the <bvname/> or <exname/> elements. In other words, any <bvname/> or <exname/> elements appearing immediately subsidiary to <classif> are considered to be food names. If desired, <cmt/> elements can be used with either of these elements to identify a particular type of name being used. Similarly, food groups and codes used in particular countries or regions are reported through the <specific classification> elements. Food names or groups may also be reflected in the <srcdbid> element, depending on how the original data base is organized.

<obtain-area/> Area or country where the food is purchased. The content of this element should preferably be an "Alpha-2" country identifier chosen from ISO 3166; see the discussion of the <country> element for further information.

Section C: Description of "single" foods

<origin/> The food source as a whole animal or plant.
<taxonm/> Taxonomic or scientific name. The content is unformatted text.
<variety/> Variety of source (origin).
<part/> Part of plant or animal.
<origin-area/> Country or area of origin. See <obtain-area/>, above; the two contents should be identical in format and usage.
<manfctr/> Identification of the manufacturer of a food. The content should contain only elements, one of which is <addr/> and another of which may be <batch>, with no end-tag, to list the batch or lot number.
<ingred/> Listing of ingredients as indicated for the description system. The different ingredients should be separated by the special delimiter <->. These are "minor ingredients" if the conventions of this section apply, i.e., if food source and scientific name are listed. Or, if the food is "multi-ingredient" (section D), the content of <ingred/> may consist of either unformatted text or elements such as those in this section to describe each ingredient.
<processing/> Processing or preparation information, including locations, listed using the open-ended keyword system.
<preserv/> Preservation method, using the open-ended keyword system.
<degree-cook/> Degree of cooking, using the open-ended keyword system.
<cond-prod/> Conditions of production.
<matur/> Maturity or ripeness, using the open-ended keyword system.
<stor/> Storage conditions, including length of storage, etc.
<grade/> Grade of food.
<contain/> Container or contact surface.
<phystate/> Physical state, shape, or form.
<colour/> Colour of food.
<photoref/> Location of photograph. The <ref/> element should be used subsidiary to this one if it is appropriate to refer to published material.

Section D: Description of "mixed" foods

<ingred/> Ingredients, see above.
<recipe-proced/> Recipe procedure.

<recipe-place/> Place processed or prepared.
<photoref/> Location of photograph, as discussed above.
<manfctr/> Identification of the manufacturer of a food. As above.
<contain/> Container or contact surface, as above.
<preserv/>Preservation method, as above.
<stor/> Storage conditions, as above.
<fnlprep/> Final preparation.

Section E: Customary uses of food

<portion/> Information about typical portion. See also <meas/>. Unlike the <meas/> element, the description system calls for identifying the portion, not just classifying it into a small number of categories.
<avail/> Availability, frequency, and season of consumption.
<dietplace/> Place in diet.
<fduser/> Food users.
<fdpurpose/> Purpose of food.

Section F: Sampling and laboratory handling of food

<smpldate/> Date of collection. The format should be identical to that for <date>.
<smplwght/> Weight of sample.
<edible/> Description of edible portion. See also <meas/>.
<refuse/> Description of refuse. See also <meas/> and <refuse/>. This is an example of the same generic identifier defining a different element (with different content) because it appears in a different context.
<smplcollect/> Place of collection.
<smplhand/> Handling of sample. Content consists of elements, each of which requires both a start-tag and an end-tag: <suppl-lab/> (between supplier and laboratory), <lab-arvl/> (on arrival at laboratory), <lab-strg/> (laboratory storage and handling).
<anal-stray/> Strategy for analysis.
<anal-ran/> Reason for performing analysis.

Example

```
<classif>
<bvname> Fried calf liver </bvname> <obtain-area/> au </obtain-area/>
<origin/> beef </origin/> <taxonm/> bos taurus </taxonm/>
<processing/> raw </processing/>
<cond-prod/> free range </cond-prod/>
<smpldate/> 1990.03.15 </smpldate/>
</classif>
```

This example is derived from several of the partial examples provided by Truswell et al. [32]

<fddflt>

The <fddflt> element is an immediate, and structural, subsidiary of the <food> element. It prescribes defaults for various subelements associated with a particular food, especially quantities such as common measures.

Description

Both start-tag and end-tag are required. The content is precisely the same as that for the <comp> item.

Format

The content of <fddflt> has precisely the same structure as a <comp> element (i.e., that of a <specific component>), except that all subsidiary elements are optional; the <fddflt> itself must be omitted if all immediately subsidiary elements are omitted. A data value occurring in a subsidiary of <fddflt> is determined for the entire food and the corresponding value must not occur in the same position in any other subsidiary element of the same <food>. In addition, it must not correspond to a value in the <dflt> element.

<Fddflt> is most often used to specify that a food is being reported in common household measures or as-purchased quantities, e.g., with <meas/> as its significant subelement.

The details of application and use of this element are discussed in Chapter 3.

Example

```
<food>
<classif> classification elements </classif>
<fddflt>
<meas/> fruit </meas/>
</fddflt>
```

specific component information

The above example would imply that all values for this particular food are "per fruit", rather than "per 100 grams edible portion". To specify both "per 100 grams edible portion" and "per fruit" as separate entries, the special delimiter <-> would be used both with this element and in the various specific component elements, e.g.,

```
<food>
<classif>
<ifri> .... </ifri>
<usda...> .... </usda...>
</classif>
<fddflt> * <-> <meas/> piece <cmt/> one fruit </cmt/> </meas/> </fddflt>
<comp>
<water> 35 <-> 7 </water>
```

```
<fat> 10 <-> 2.25 c/fat>
</comp>
</food>
```

The above is equivalent to writing:

```
<food>
<classif>
<ifri> .... </ifri>
<usda...> .... </usda...>
</classif>
<comp>
<water> 35 <->
7 <meas/> piece <cmt/> one fruit </cmt/> </meas/> </water> <fat> 10 <->
2.25 <meas/> piece <cmt/> one fruit </cmt/> </meas/> </fat>
</comp>
</food>
```

<specific component>

Each <*specific component*> element is an optional, immediate subsidiary of a <comp> block. It contains data about one component with respect to one food. <Specific component> is a placeholder for, and incorporates by reference, all of the food component identifiers specified in Chapter 2 of *Identification of Food Components for INFOODS Data Interchange* [17], as well as additional food component "tagnames" which might be registered in the future.

The term <specific component> is shown in italics as a reminder that it is not an actual tag and never appears in an interchange file but, instead, is a placeholder for a series of individual elements.

Description

Both start-tag and end-tag are required. The content of <*specific component*> generally consists of one required formatted data item (the data value, a numeral or a "missing" or "trace" indication) and any associated information (which may be component-specific numerals, keywords, and/or subsidiary elements), along with optional immediate subsidiaries selected from the following extensible list (the numbers in parentheses indicate the page numbers on which the element descriptions begin):

<unit/> (88)	<meas/> (89)
<srcfri/> (94)	<srcorg/>

and the elements listed in this document as <*data description*>.

This single-data-value-and-associated-information content block may be repeated in form, with separate blocks separated by the special tag <->.

The actual content of a particular <*specific component*> will be as specified in the element's registered description, as maintained by the appropriate registration authority, but will always include the list above, possibly extended by the registration authority for this "generic element".

Format

Each <*specific component*> contains data about a particular component of a particular food (that food whose data is in the <food> element to which this element is subsidiary).

More than one measurement may be available for a single component of a single food record. Typically, the second or subsequent measurements would represent different statistical estimates (e.g., a median rather than a mean) or different reference quantities (e.g., household units, or food as purchased, rather than a measure per 100 grams edible portion). When this situation occurs, the entire content of the element may be repeated for each additional value, using the special tag <-> as discussed under "Repeated and Counted Elements" in Chapter 3.

If any single component's or derived component's element contains such a repeated content, then all of the components and derived components should contain the same number of repeated contents. There is one exception to this rule: if fewer contents appear for some components than others, the missing ones will be treated as if they were present but empty (e.g., "<a> 1 <-> 2 </a>" and "<a> 1 <-> 2 <-> <-> </a>" are treated as equivalent). The empty or missing repetitions have the same significance that would be implied if the element for the component were completely omitted (there was no other value set being reported for the food in question).

As mentioned above, the actual content of any particular <specific portent> will be as specified in that element's registered description, as maintained by the appropriate registration authority. However, the content should be thought of as consisting of two parts: (i) the required data item and any food-component-specific numerals, keywords, and/or subsidiary elements that are needed to qualify it, and (ii) the collection of optional elements that specify non-standard units or quantities measured, and statistical and sampling descriptions. This second part is optional for all <specific component> elements, and is not listed with their registered descriptions. The list of such subsidiary elements given above may be extended by its registration authority to provide for additional per-component information, but a program processing the interchange file may ignore any of the second-part elements which it does not recognize.

<specific derived component>

Each <specific derived component> element is an optional, immediate subsidiary of a <drvd-comp> block. It contains data about one component with respect to one food. *<Specific derived component>* is a placeholder for, and incorporates by reference, all of the food component identifiers specified in Chapter 3 of *Identification of Food Components for INFOODS Data Interchange* [17], as well as additional derived food component "tagnames" which might be registered in the future.

The term <specific derived component> is shown in italics as a reminder that it never appears in an interchange file but, instead, is a placeholder for a series of individual elements.

Description

Both start-tag and end-tag are required. The content of *<specific derived component>* generally consists of one required formatted data item (the data value, a numeral) and any associated information (which may be component-specific numerals, keywords, and/or subsidiary elements), along with optional immediate subsidiaries selected from the <srcfri/> or <srcorg/> elements, applicable *<data description>* elements, and any additional elements that may be registered in the future.

<Data description>, as used here, refers to the collection of generic identifiers that can be used to describe the statistical and related properties of the data value, e.g., what statistic is being reported, how the value is distributed, and any available information about accuracy or precision.

This single-data-value-and-associated-information content block may be repeated in form, with separate blocks separated by the special tag <->.

The actual content of a specific *<specific derived component>* will be as specified in the element's registered description, as maintained by the appropriate registration authority, but will always include the list above, possibly extended by the registration authority for this "generic element".

Format

Each *<specific derived component>* contains data about a particular derived component of a particular food (that food whose data is in the <food> element to which this element is subsidiary).

More than one measurement may be available for a single component of a single food record. Typically, the second or subsequent measurements would represent different statistical estimates (e.g., a regulatory minimum rather than an analysed mean). When this situation occurs, the entire content of the element may be repeated for each additional value using the special tag <-> as discussed with *<specific component>*.

In the special case of a repetition to permit <*specific derived component*> values for different units or measurement quantities, the corresponding <*specific derived component*> repetitions need only be given once, with the remaining (counting from the left) empty omitted repetitions assumed to have the same value. This is because derived components have "units" that do not involve the raw quantities sampled or the raw component amounts determined by experiment.

The <unit/> and <meas/> subelements are prohibited as subsidiaries here for the same reason.

As mentioned above, the actual content of any particular <*specific derived component*> will be as specified in that element's registered description, as maintained by the appropriate registration authority. However, the content should be thought of as consisting of two parts: (i) the required data item and any food-component-specific numerals, keywords, and/or subsidiary elements that are needed to qualify it, and (ii) the collection of optional elements that specify statistical and sampling descriptions. This second part is optional for all <*specific derived component*> elements, and is not listed with their registered descriptions. The list of such subsidiary elements given above may be extended by its registration authority to provide for additional per-component information, but a program processing the interchange file may ignore any of the second-part elements which it does not recognize.

6. Data values and data description

INTRODUCTION

This chapter describes the interchange elements that are used to describe the data values themselves: units of measure, statistical values, and their interpretation, using the same organization found in the previous two chapters.

<unit/>

The <unit/> element is an optional immediate subsidiary of the various <*specific component*> elements. It specifies non-standard units for the data value being reported.

Description

Both start-tag and end-tag are required. The content consists of a keyword chosen from the list below and subsequent registrations.

Format

The unformatted data item which is the content of the <unit/> element is usually a prescribed keyword, but may be a descriptive word or short phrase. It describes the numerator of the "unit" of the value of a <*specific component*>*,* such as "milligrams" or "ounces". Compare with <meas/>, which, in the same sense, expresses the denominator.

The <unit/> element may occur immediately subsidiary to a <*specific component*> only if it does not occur within the <fddflt> element of the same enclosing <food> and does not occur within the interchange file's <dflt> element. If it occurs nowhere, then the units specified in the registered definition of the containing <*specific component*> or <*specific derived component*> are assumed. The <unit/> element should not be used when the values reflect the default units for the particular food component.

Keywords

The content of the <unit/> element is a keyword value. The initial set of keywords are as follows. This list can be expanded by future registrations.

KEYWORD	DEFINITION
g	Grams
mg	Milligrams
mcg	Micrograms
mmol	Millimoles
l	Liters
ml	Milliliters, cubic centimeters
mcl	Microliters

Example

```
<ca> 3.0 <unit/> g </unit/> </ca>
```

<meas/>

The <meas/> element is an optional immediate subsidiary of each <*specific component*>. It specifies a non-standard measure of food for which a quantity of the component is being reported.

Description

Both start-tag and end-tag are required. The content of a <meas/> element is a keyword, or keyword and element, chosen from the list below, optionally followed by a <cmt/> element.

While this element may be used to identify non-standard data for interchange purposes, data base compilers should be aware that quantities other than the conventional "per 100g edible portion" are likely to be extremely difficult to interpret in international or most comparative contexts. Consequently, this element should, if possible, be used only to identify supplemental data, e.g., values for specific household quantities or portions in addition to the "per 100g" quantities, not instead of them.

Format

The content of the <meas/> element is usually a prescribed keyword, but some keywords require additional description, as shown below. The element describes the denominator of the "unit" of the value of a <*specific component*>, such as "per 100 grams" or "per one fruit" (but the word "per" is never included). Compare with <unit/>.

The <meas/> element may occur immediately subsidiary to a <*specific component*> only if it does not occur within the <fddflt> element of the same enclosing <food> and does not occur within the interchange file's <dflt> element. If it occurs nowhere, then the preferred measure "(per) 100 grams edible portion" is assumed. The <meas/> element should not be specified when the measure in use is the default.

Keywords and Structure

The content of the <meas/> element is a keyword value. The initial set of keywords is as follows. This list can be expanded by future registrations.

KEYWORD	STRUCTURE AND DEFINITION
e100g	Normally the default: per 100 grams edible portion. Neither <qty/> nor <refuse/> elements may be used subsidiary to this element.
t100g	Per 100g as purchased. Should be used when possible with the <refuse/> subelement, described below. The <qty/> element may not be used with this element.

piece	Per "piece", for those foods for which this is an appropriate unit. For example, this would be used to designate "per fruit" for fruits, "per chicken" for chickens, "per potato" for potatoes, and so on. So-called "household portions" are, for reporting purposes, special cases of "piece". Unless the measure is already a weight, it is unlikely to be usable outside the originating food culture unless <qty/> is also supplied, so that information should be supplied if at all possible.

The absence of a <refuse/> element will be taken by most receiving parties as indicating that the values are supplied without refuse, i.e., as indicating that values supplied with "t100g" or "piece" are completely edible. Consequently, a <refuse/> element should be supplied if this is not the case. If it is not possible to supply a <refuse/> element when the quantity is "as purchased" or otherwise contains an inedible fraction, a <cmt/> element should be included that indicates this.

When the food is expressed in unconventional "household measures" or "as purchased", it is important to remember that people receiving a data file may be unfamiliar with the food. If information is available, the <qty/> and/or <refuse/> elements should be used to provide the basis for an approximate conversion to 100 grams edible portion if that is required, and additional information should be provided, as part of the food description, that will better describe portion sizes, etc.

Unfortunately, there has been little research or standardization in the description of this critical area of food quantity description. It is likely that, at least in the near future, food composition data bases that include measures other than "per 100 grams edible portion" will require extensive textual description, in <cmt/> elements associated with <meas/> and in food description elements to make the values useful and comparable outside, and possibly inside, the country of origin.

Subelements

<qty/> Used to provide an estimate of the quantity, in grams, associated with a household or "as purchased" portion as eaten.

<refuse/> Used to provide an estimate of the fraction of a food "as purchased" that will be lost in conversion to "portion as eaten".

These two elements are ideally used together when "as purchased" quantities are involved. The second provides the conversion between quantity or portion "as purchased" and the amount considered edible and the first provides the conversion between that quantity and the specified portion size. The specific relationship is that, if the <qty/> and <refuse/> values are respectively Q and K, then edible portion size in grams = Q and the quantity purchased in grams, P, is

P= Q / (1 -R)

The information is often hard to obtain and quantify, and both of these elements are optional. Additional information may be provided as part of the food description subsidiary to <classif>.

Examples

```
<meas/> piece <cmt/> one fruit </cmt/>
<qty/> 100 <cmt/> seeds eaten, no refuse </cmt/>
</qty/>
<meas/>

<meas/> piece <cmt/> half-pound steak </cmt/>
<refuse/> 030 <cmt/> trim fat and bone </cmt/> </refuse/>
<qty/> 315
<cmt/> half-pound less refuse, large variation in practice </cmt/>
</qty/>
</meas/>
```

<qty/>

The <qty/> element is an optional immediate subsidiary of the <meas/> element, used with some of the keywords for that element. It specifies an approximate conversion between a household portion or quantity as purchased and grams as prepared or consumed. Its exact semantics depend on the keyword of the <meas/> element with which it is used.

Description

Both start-tag and end-tag are required. The content consists of a numeral, typically representing the number of grams in the portion, piece, or size of product.

Unless the measure associated with the <qty/> element represents an absolute unit (e.g., "per ounce as eaten") the quantity divisor will represent a value that varies. For example, if the <meas/> element contains "piece" and a <cmt/> element indicating "one fruit", <qty/>, as described here, will represent the weight of an average fruit. As food composition data improve, it will probably be desirable to associate the same types of statistical distribution information about this value as are used to describe the data values themselves. This element will be extended as needed to accommodate that information.

Format

The content of a <qty/> is a numeral, in floating-point notation if needed. It will be used as a unit-free divisor. The content may optionally contain a <cmt/> element, and additional elements may be added as discussed above.

<refuse/>

The <refuse/> element is an optional immediate subsidiary of the <meas/> element, used to supplement "as purchased" quantities. It is expressed as a fraction of the total amount that is refuse, with an optional <cmt/> that describes the part that is discarded.

Description

Both start-tag and end-tag are required. The refuse element is used to describe the amount of refuse, or waste, in converting between the portion of a food as purchased (or otherwise obtained, as in "raw" form) and the edible portion of the food.

For example, if the <meas/> element contains "piece" and a <cmt/> element indicating "one fruit", <refuse/>, as described here, might represent the weight of the pit and inedible peel (as discussed under <meas/>, <cmt/> and food description elements that indicate what is considered edible are critical for understanding the data).

For natural products, the actual refuse removed prior to consumption will typically differ from one example to the next. Consequently, a <refuse/> value always represents, at best, an average value with a potential, but rarely well-understood, variance.

As food composition data improve, it will probably be desirable to associate the same types of statistical distribution information about this value as are used to describe the data values themselves. This element will be extended as needed to accommodate that information.

Format

The content of a <refuse/> element consists of a numeral, representing the fraction of the product that will be discarded, and an optional <cmt/> element describing the part that is discarded.

<srcfri/>

The <srcfri/> and c srcorg/> elements are immediate subsidiaries of the <*specific component*> or <*specific derived component*> cements. They may be specified with <fddflt> when the same data records are used for all component values for a given food; in this case, no c srcfri/> or <srcorg/> elements should appear in the food record itself.

When data values are calculated or otherwise derived from values in other tables, <srcfri/> is used to list the international food record identifiers of the data records that contributed to the calculations. <Srcorg/>, by contrast, is used to keep track of different unpublished data sources assembled by the compiler for the same food. For example, if two or more laboratories were used for different values, c srcorg/> could be used to identify the laboratories with the food components they supplied.

These elements are part of the overall system of tracking the use and evolution of data values discussed with the <ifri> element. In general, <srcfri/> will be used for published data values that have already entered the INFOODS interchange environment or that of one of the regions, so that a food record identifier has been assigned, and <srcorg/> will be used for other data values.

With one exception, listing more than one <srcfri/>, or both <srcorg/> and <srcfri/> elements, for a single food component will be rare. The exception arises when multiple sources, such as values from several tables, are used to derived a single value to be published.

"Srcfri" may be thought of as an abbreviation for "source food record identifier"..

Description

Both start-tag and end-tag are required. The content consists of an international food record identifier. If more than one is required to identify the data source, more than one <srcfri/> element may appear.

Format

The content of <srcfri/> consists of one or more unformatted strings, each of which is an international food record identifier. The strings are separated by the <-> delimiter.

<srcorg/>

The <srcorg/> element is an immediate subsidiary of the <*specific component*> or <*specific derived component*> elements. It may be, and often will be, specified using <fddflt>. This element is part of the overall system of tracking the use and evolution of data values. Its specific application is discussed with <srcfri/>. "Srcorg" may be thought of as an abbreviation for "source food record origin".

Description

Both start-tag and end-tag are, required. The content consists of a set of elements representing an original data source of data not included in the international food record identifier system. Several <srcorg/> elements may appear if needed. The element sets may contain <ref/> elements, typically to identify published articles that contain data, or <cmt/> elements for more local information, e.g., laboratory identification. The list of permitted elements that may appear subsidiary to this one may be expanded as requirements become obvious.

With one exception, listing more than one <srcorg/> element, or both <srcorg/> and <srcfri/> elements, for a single food component will be rare. The exception arises when multiple sources, such as values from several laboratories, are used to derived a single value to be published.

Format

The content of <srcorg/> consists of a set of elements, which represent an internal data source. If there are multiple sources, more than one <srcorg/> element may appear, as discussed above.

<data description>

The <*data description*> elements are immediate subsidiaries of the various <*specific component*> and <*specific derived component*> elements. They specify various statistical properties of the data value or sampling information relevant to the particular food component.

The term <data description> is shown in italics as a reminder that it is not an actual tag and never appears in an interchange file but, instead, is a placeholder for a series of individual elements.

Description

The content and structure of the various <*data description*> elements are different.

Format

The content of the specific <*data description*> elements will be as registered.

Philosophy and Categories

The data description elements are intended to provide methods of supplying and identifying both data about food components and "metadata" (descriptive information about the data and conventional statistics), which are additional values and description about those data. The categories immediately following are influenced by both general data classification theory and the realities of practice in handling and presenting food composition data. We use five major categories, two of data (or statistics derived from the data) and three of metadata. It is possible to think of the <unit/> element as part of this group as well.

The intent of the discussion, and the elements specified here, is simultaneously to provide a framework for extensive description of data, statistical and otherwise, and to provide some justification for doing this. As with other components of the interchange system, none of these elements is required for minimal interchange. Even with more extensively documented data bases, the elements should not be used unless the data associated with them are available. Indeed, some elements are provided to identify descriptive values that appear in food tables that INFOODS does not recommend including. The maintainer of a table or data base that contains only point estimates of values which can be treated as means can ignore these elements entirely.

The element groups are listed below. The first two of these would normally be considered as statistics (or "data") and the other three provide metadata, including information about relationships to, and among the data. Each of these is described in more detail after the listing.

1) A "best" estimate of location in the intuitive, rather than formal statistical, sense of "best".

2) Statistics, each as part of an element that indicates what sort of statistic it is. The use of the term "each" should not be construed to imply that these are necessarily single values. A list of the data is a valid set of statistics for those data.

3) The treatment (i.e., processing or "data cleaning" operations) used to transform the data into the set of values on which the statistics actually report.

4) A description of the distribution itself. To some extent, descriptions of distributions represent empirical beliefs about the data that influence the choice and application of treatment procedures, as immediately above, so this category could also be described as "a structured description of beliefs about the data".

5) A description of beliefs about the distribution. Beliefs about the data that are more general or, especially, qualitative or subjective than those that can be summarized by distribution statistics or summaries would typically fall into this category.

Consistent with the general interchange model, any of these categories for which information is not available may be omitted. Nonetheless, it is useful to understand that each of the categories, with the possible exception of the first, always exists at some level, however trivial or embedded in the subconscious of the table-preparer. No known food composition table or data base at present specifically contains either of the two last categories. The third normally appears only in the paper archival files that record the progress of data from laboratory to the file of raw data and from there to more refined and "cleaned up" data bases.

These categories are inexorably intertwined with each other and with <unit/>, both in logic and in how they are to be handled in the interchange system. If different location estimates or statistics are associated with, e.g., different units of expression or different data cleaning procedures, or different sets of beliefs, then a separate group of values (statistics and metadata) must appear for the alternative units or cleaning procedures.

Earlier sections of this document have discussed the use of the special delimiter element <-> to divide repeating groups of information. <-> may appear immediately subsidiary to a <*specific component*> or <*specific derived component*> element to indicate that it contains two or more groups of data. When "<->" is used in this context, we describe these groups as "statistical data groups". Each of these may have its own values, its own units, and its own statistical description of the data values. No mechanism parallelling <dflt> or <fddflt> is now defined or contemplated to permit implied copying of values from one of these groups to another, or for using one such group to establish defaults for another (which is much the same thing). If two groups of data are needed, and some of the information overlaps, then it must be duplicated.

Because of the nature of the best estimate of location, one group can contain at most one of these. If more than one such estimate is provided, then there must be a corresponding number of <data descriptions>, presumably with different applicable <unit/> or <meas/> elements associated with them or with different data treatments (e.g., methods for cleaning or rejecting outliers).

For ease in processing at the receiver end of an interchange, data producers should be encouraged to place the most representative or most internationally useful values and statistical information first when they have opinions on which is most representative or useful, but this is not a rule of the interchange system. For example, if a food composition table being translated into interchange form contains both data on the basis of 100 grams of edible portion of a food and data on the basis of the food as purchased, we would strongly recommend that the data should be reported in that order for each nutrient.

THE SETS OF VALUES

Two categories of data and three of metadata are listed above. This section explains those categories, and can safely be skipped by any reader who has an adequate grasp of the categories from the brief discussion above.

1) Best estimate of location

This category represents several realities, rather than statistical purity, although there is analogous statistical terminology for it (below). Many food composition tables report only a single value for a food. Most other tables feature one value prominently. In all of these cases, that value represents the value that a table-producer might supply in response to the question, If I have to use a single value to represent the amount of this particular nutrient in that particular food, what should I use? That value may be true or false, representative or misleading, but it presumably represents the best estimate available to that food table producer at that time. It is often inappropriate to label it, for example, an "average value", since that term has a very precise meaning and, in some cases, averages may be known to the table producer to be inappropriate. The interchange system makes provision for precisely identifying a value as the mean when that is appropriate.

We would expect that some variety of location estimate (possibly a "trace" indication, rather than a number) would appear for substantially every food component that is reported. Outside this section, examples of fragments of interchange files consequently omit specific location-statistic-identifying information.

2) Statistics about the data

As discussed immediately above, it is often inappropriate to identify many of the values that appear in food tables with precise statistical terms, such as "mean". At the same time, when descriptive statistical values are available, the interchange system must support reporting them, and reporting as much detail about them and their derivation as can be found. This general category subsumes the statistics themselves; the next one begins the description of the derivation of the statistics.

In principle, any [sample] statistic about the data may appear in this category. Examples of such statistics, each of which would be represented as one or more values tagged to indicate the statistic it represents, would include estimates of central tendency such as the mean, the median, and the geometric mean; estimates of spread such as the variance, standard deviation, range (expressed as the difference between maximum and minimum values or as the maximimum and minimum values themselves), and hingespread (difference between the upper and lower quartiles); critical values such as the maximum and minimum' 15th or 85th percentiles, or an 80% confidence limit; estimates of accuracy, such as the standard error or non-parametric estimates of standard error such as the jackknife; the sample size; and so forth.

This category also includes a special element that identifies the particular statistic represented by the best estimate of location if, in fact, that represents a precisely defined statistic. This provides a compact notation and prevents giving the appearance of more information than is actually present.

While the term is often too vague to be of significant use, this category includes all or most of what are often referred to as "descriptive statistics".

3) Treatment of the data

It is typical with observed or estimated data in general, and it seems particularly true of food composition data, that one rarely takes unevaluated "raw>' data, computes, e.g., a mean, and reports the value. Indeed, with most food composition data, such an approach would be irresponsible, as Greenfield and Southgate [11] argue most forcefully.

Instead, one should, and does, evaluate, removing values that are obviously bad, and potentially adjusting others because of what is known about the characteristics of the particular food sample. The evaluation process invariably involves the application of experience with, and theory about, food composition to the data, and may involve the application of formal procedures based on statistical theory.

With most food composition tables in the past, the process of data evaluation and treatment has not been described in detail, partially because of a belief that no one was interested or would be able to make use of the information, and partially because these was no framework in which to describe it. Consistent with the goal of organizing the interchange system so that an interchange file can be self-contained and include all information that is available, we wish to define a framework in which the information about data evaluation and treatment can be reported if that is desired.

The elements in this category describe the treatment itself, and the next two categories are available to describe the assumptions on which the treatment is based if those are relevant and available. Since choices of treatment used will often lead to different statistical values, there is an inexorable relationship between these elements and the statistical ones (categories 1 and 2) discussed above. As a corollary, if treatment elements are omitted, the receiver of the data will usually infer that the treatments applied are "safe" or "transparent" and do not affect the reported data values. Just as the descriptive values can include either formal statistical procedures or subjective or objective decisions based on experience and examination, the treatments can also. The content of this category could then include statements with such meanings as "examined the data on the basis of experience and discarded obviously silly values", or "applied a ten percent trimming rule to the sample data to eliminate erratic behavior in the tails of the distribution", or "discarded measured cholesterol values for this plant product".

This category will rarely appear without at least some statistics about the data (category 2), but there are exceptions. In particular, a data treatment statement such as the last one above would justify reporting a best estimate of location of zero, even if there were no specific descriptive statistics reported, and even if the measurement showed a trace quantity. Cholesterol values for plant products are a traditional example of this type of situation.

Nothing here is intended to encourage or deprecate any particular procedure, decision, or decision-making process. Instead, as with other aspects of the interchange system, it is important to facilitate the transfer of whatever information is available about what was done so that the data recipient can adequately perform his or her own critical evaluation of procedures and descriptions against the background of the use for which the data are intended. Consequently, whether or not it is reasonable to discard data that seem to indicate cholesterol in plant products is not at issue here; what is at issue is the degree to which the interchange system facilitates the reporting of such a decision, if it was made, and, to some extent, the degree to which it encourages the reporting of such decisions.

On the other hand, when one is going to report information about the properties and distribution of data, some statistics are clearly better than others. Some comments on that subject can be found in the descriptions of the individual elements, others appear as part of the INFOODS recommendations on compiling food composition tables [24].

One should also avoid the temptation to believe that data evaluation and cleaning methods based on statistical procedures are "better" than those that involve proportionately more of the wisdom and sophistication of an experienced scientist. The opposite may be true: with data as complex as most food composition data become by the time they are reported in a table, there may not only be no reasonable substitute for the judgement of a scientist, but the uncritical application of a statistical procedure may be appropriate only when neither experience nor theory is available.

4) Description of the distribution

It is sometimes possible to describe a sample distribution in ways that move beyond summary statistics. The most obvious of these is a simple listing of data points, or a listing of the frequencies or cumulative frequencies of groups of data points, with the groups determined by either equal-interval categories or some other rule, or a listing of the values associated with certain selected fractions of the data (collections of percentiles and values at the first and second standard deviation points fall into this category). To some degree, this information, if provided, supplies additional empirical information about the degree to which the summary statistics can be believed to be useful (if the best estimate of location is not a particular summary statistic and reported as such, one's confidence in it is simply one's confidence in the ability of the analyst and the table compiler to identify a "best" value; while this sounds dangerous and subjective on first glance, it reflects what is actually done and is a reasonable approach).

5) Beliefs about the distribution

Especially when statistical procedures are applied to evaluate or clean data, those procedures are typically based on beliefs about the underlying distribution and what values are, and are not, "possible". Beliefs could include such assertions as "I know the underlying distribution is normal (or exponential, or...)", "I know that there is probably normality in the population distribution, but the instrumentation is unable to detect concentrations below 0.0001 percent, so the sample distribution will be censored in the left tail", "There is reason to believe that the sample distribution is a mixture of two populations, so the data may be multi-modal", "I know that cholesterol does not appear in plants", and so forth. It is useful to document these assumptions and statements about beliefs because they may be controversial. Scientists who would agree should know that their assumptions are shared; those who disagree should be able to evaluate the data accordingly.

This information is typically even more abstract than the description of the distribution, and we would not expect it to be reported very often. At the same time, as discussed above, there is merit in arranging for it to be reported if it is available.

Common Practice and Recommendations

As in its other sections, this chapter is devoted to providing ways in which food composition data can be structured and identified for interchange, and possibly other, purposes. At the same time, this chapter probably contains more unfamiliar terminology, and elements describing unfamiliar concepts, relative to common practices with food composition data, than most others. The most common practice has been to report only a single value for each nutrient for a given food (which a statistician might call a "point estimate of location"). Less common, but still popular, has been to report the point estimate along with some indication of how much the actual value might be expected to vary-typically minimum and maximum values, a confidence interval, or a standard error, together with a sample size.

Since many existing tables and data bases take this approach, the interchange system partially provides for it by including an inexactly described point estimate of location. It also provides for upper and lower bounds (<bounds>), standard errors (<serr>), and sample size (<smsz>). However, when some of these estimates are calculated from the very small samples typically encountered in food composition work (often as small as five or six or fewer), they tend to be very sensitive to extreme values (see Rand et al. [24] or Rand [25] for more discussion of this subject in a food composition context and any of several standard statistical references-e.g., [8, 28, 34]-for a more extensive statistical treatment).

There are two current trends in statistical data analysis and data description that can be thought of as addressing the issues of better describing and understanding small samples of potentially guise irregular (e.g., non-Gaussian) data. One of these concentrates on the use of more "robust" estimates (those that are less prone to be significantly distorted by a few extreme, or otherwise bad, data points) and sometimes on describing the sample without trying to make population inferences. The other involves the explicit combination of external data or knowledge with the sample data to provide more information about both. The treatment in this chapter is intended to support both of those points of view, either of which is probably preferable to the traditional approaches unless sample sizes are quite large. While we hope that future INFOODS recommendations will address these issues in more detail, perhaps the best explanation of the first approach as it is applied here (although by no means an introductory tutorial) is provided by Hoaglin et al. [14]. For the second approach, we recommend the discussion by Efron and Morris [7] for the empirical approach or Howsen and Urbach [15] for the broader philosophical issues involved.

STRUCTURE OF THE ACTUAL ELEMENTS

Within a <*data description*>, the best estimate of location must appear first, if it is to appear at all. It is important to note that it is data in the <comp> element, and is not part of some other element (ignoring the SGML interpretation of the special delimiter <->).

The other four categories of statistics and metadata may appear in any order within the <*data description*>. These categories are not represented as data within the content of the <comp> or <drvd-comp> elements, but as [tagged] elements, as specified below.

The current availability of data is such that we need not define the third through fifth categories in great detail at this time. We must be prepared to do so when they are needed, and must be reasonably assured that they can be accommodated without doing violence to the interchange system. Consequently, just as we have left "component content" somewhat undefined up to this point, it is now appropriate not to try to define it completely (which could be a never-ending task) but to define appropriate structures and then wait for actual practice to require further definition.

Consequently, having described what we mean by a <*data description*> we define each one as consisting of the following:

1) The best estimate of location, as described in the documents that define the food component tagnames. This estimate may contain elements in its content, as specified in those documents. It is optional, but we would expect that it will almost always appear.

2a) The descriptive statistics, each identified by a tag that indicates what it is. The elements will typically not require end-tags but will consist entirely of one or more numeric values; consequently the generic identifiers will not end in slashes. There will be no tag or element whose function is to delimit the descriptive statistics from the other content of the <*data description*>.

2b) As a convenience, to avoid the appearance of more information than exists, and to facilitate the specification of defaults that apply to an entire interchange file, an additional tag is introduced, named <loctype/>. Its purpose is to designate the location statistic actually represented by "best estimate of location", when that value can be responsibly reported as a location statistic. Its content is a keyword from a restricted vocabulary, which should be the list of tagnames for location statistics.

The <loctype> element can be mixed with the elements that describe particular location statistics, with slightly different meaning. "<NA> 5.2 <loctype> MEAN </NA>" indicates that there are 5.2 mg of sodium, that this is the best estimate of location, and that the value is the mean. We would usually take " <NA> 5.2 </NA> " to be identical to this, but the longer form provides slightly more confidence. In a slightly more elaborate form, or with a more devoted data base or interchange file producer, we might encounter

<NA> 5.2 <loctype> mean <median> 5.1 </NA> or
<NA> 5.2 <median> 5.1 </NA>

with the same significance as the examples above, but indicating that the median was also computed and is provided. There is, however, another case, with slightly different semantics, which should not appear unless the data base developer has determined that there is a need to make a very specific point:

<NA> <mean> 5.2 <median> 5.1 </NA>

reports a mean and median as above, but suggests that the data base developer or maintainer is quite explicitly not willing to make an assertion as to which of these is the best estimate of location. Such an unusual assertion should normally be accompanied by an explanatory comment.

An obvious alternative would simply be to require table compilers to specify the statistic associated with the best element of location, with a possible value implying "unknown". There are several reasons why this was avoided. This estimate may be somewhat subjective or intuitive, rather than a well-defined statistic. Second, especially when recording tables compiled years ago, the precise statistical parameter would often be unknown, at least without further qualification, and the interchange system should not encourage people to guess at information that does not actually exist.

3) The description of the treatment or processing. Future work may be required to supplement whatever free text is used by recognizable and comparable tags or keywords. However, since, as far as we know, information of this type has not yet appeared in any food composition table (although it is becoming prevalent in other fields), free text description should be used for the near future. The <sclean/> tag identifies this information.

4) The empirical description of the distribution. The <edistr/> element identifies this information.

5) The description of more subjective beliefs about the distribution. As with <sclean/>, there is now provision only for free text content, but work should be started on specific elements and keywords as soon as that is practical. This information is identified with the <sdistr/> tag , since it provides a subjective description of the [hypothesized] population distribution.

<loctype>

The <loctype> element is an immediate subsidiary of <*specific component*> elements. More specifically, it is a component of <*data description*> and immediately follows the data value associated with <*specific component*> or <*specific derived component*> elements. It is used to specify the exact meaning of the "best estimate of location" for its associated food component (i.e., the <comp> or <drvd-comp> element to which it is subsidiary). "Loctype" may be thought of as an abbreviation for "location type" or "type of location estimate".

Description

Only the start-tag is permitted. The content consists of an unformatted string (whose first, and usually only, "word" is a keyword) and terminates when another tag is encountered. The keywords represent names of location statistics. If this element is omitted, most data base users will infer that the food component value represents a mean value. Use of <loctype> with "mean" reinforces and confirms that belief when that is appropriate; use of <loctype> with another value indicates that the mean was not considered to be the best estimate of location.

Format

The content of <loctype> consists of one member of the following list. The keywords in this list correspond exactly to the elements for specific location statistics that appear starting on the next page; the two lists will be expanded in parallel, and the correspondence between the generic identifier for the location elements and the keyword below, and between qualifying parts of the element and the additional information below, will be preserved.

KEYWORD
mean
median
tmean N M
locpctl N

Example

<comp> <ash> 0.69 <loctype> mean </ash>
<enerc> 354.027 FDS </enerc>
<procnt> 17.270 USDA 6.38 c loctype> mean </procnt> </comp>

The use of <procnt> here illustrates a case in which the primary information associated with a generic identifier is more complex than a single numeric value. As mentioned in the first paragraph above, the data description information follows all of the information that is directly associated with the <comp> "tagname" [17]. For <procnt>, this information consists of three values: the estimate of location, a keyword that specifies the source of the conversion factor used, and the actual conversion factor. " <Loctype> mean" specifies that 17.270 is the mean, not 6.38, which is just a conversion factor.

<mean>

The <mean> element is an immediate subsidiary of <*specific component*> elements. More specifically, it is a component of <*data description*> and follows the data value (and any <loctype> element, if present) associated with <*specific component*> or <*specific derived component*> elements. It is used to specify a mean value for its associated food component (<comp> or <drvd-comp> element) that is not the best estimate of location.

Description

Only the start-tag is permitted. The content consists of a single floating-point value representing the estimated or sample mean value for the component. Typically, this element would be used only if the best estimate of location differed from the mean.

Format

The content of <mean> is a single floating-point value representing the mean.

Examples

<enerc> 325 FDS <mean> 354.2 </enerc>

This would normally be interpreted as implying that the table compiler believed that the value 325 provides a better estimate of the total energy available than the actual mean. One would hope to find an associated <cmt/> element explaining this situation or a <loctype> element that explains how the value of 325 was derived. For example, we might see:

<enerc> 325 <loctype> median <mean> 354.2 </enerc>

This suggests a conclusion that the median provides a better estimate than the mean, but that the mean value is reported for comparison purposes.

<fat> 0.59 <loctype> pctl 80 <mean> 0.42 </fat>

This would normally be interpreted as implying that the table compiler wished to report a nominal 80th percentile value, 0.59, as a better estimate of the total fat present than the mean of 0.42. As above, one would hope to find an associated <cmt/> element explaining this situation.

<median>

The <median> element is an immediate subsidiary of <*specific c component*> elements. More specifically, it is a component of <*data description*> and follows the data value (and any <loctype> element, if present) associated with <*specific component*> or <*specific derived component*> elements. It is used to specify a median value associated with its food component (<comp> or <drvd-comp> element) that is not the best estimate of location. We take the median to be the midpoint in a sorted list consisting of any sample with an odd number of values and the arithmetic mean of the middle two values if the sample size is even. Other definitions should be described with a <cmt/> element.

Description

Only the start-tag is permitted. The content consists of a single floating-point value representing the estimated or sample median value for the component.

Format

The content of <median> is a single floating-point value representing the median.

Examples

<enerc> 354.2 FDS <median> 325 </enerc>
<fat> 0.59 <loctype> pctl 80 <median> 0.42 </fat>

This would normally be interpreted as implying that the table compiler wished to report a nominal 80th percentile value, 0.59, as a better estimate of the total fat present than the (possibly unknown) mean and that the median value was 0.42. One would hope to find an associated <cmt/> element explaining this situation.

<locpctl>

The <locpctl> element is an immediate subsidiary of <*specific component*> elements. More specifically, it is a component of <*data description*> and follows the data value (and any <loctype> element, if present) associated with <*specific component*> or <*specific derived component*> elements. It is used to specify a percentile-like value that is not the best estimate of location.

<Locpctl> is a somewhat dubious location statistic, not a "percentage point" value used to describe the distribution (see <pctpts>). In some countries, nutritional labels on food packaging for certain food components, are required to show not a mean value, but a value such that a certain percentage of the packages or portions will contain "at least that much" or "no more than that much" of the food component (depending on whether it is considered desirable or undesirable). This type of reporting requirement makes the actual value given at least as much a property of a manufacturer decision as one of sample data: "safety margins" may be included for possible future alterations in the recipe or to provide extra protection against accusations of non-compliance labelling.

While this type of information may be useful to the consumer, it would ideally never appear in a food composition table, since it is impossible to compare even approximately with, e.g., mean values. In the large sample case, we assume that the mean converges on the "the centre point" (the median or the 50% point), while this type of value would have its percentage point as a lower bound. However, since, for several reasons, it is not unusual for these values to appear in food composition tables, this element is provided to identify them.

"Locpctl" may be thought of as an abbreviation for "location percentile".

Description

Only the start-tag is permitted. The content consists of two floating-point values. The first represents the estimate and the second represents the percentile chosen. This element gives an estimate of location when a particular percentile value has special (e.g., regulatory) meaning to the data base compiler. A separate element, <pctpts>, should be used to list various percentage points as a means of describing the distribution of the data.

Format

The content of <locpctl> is a pair of floating-point values representing the value and the percentage point with which it is associated. The second value is expressed as a percentage, not a fraction, since that is the usual form of the specification.

Example

<fat> 0.42 <locpctl> 0.59 80 </fat>

<tmean>

The <tmean> element is an immediate subsidiary of <*specific component*> elements. More specifically, it is a component of <*data description*> and follows the data value (and any <loctype> element, if present) associated with <*specific component*> or <*specific derived component*> elements. It is used to specify a trimmed mean value that is not the best estimate of location. "Tmean" may be thought of as an abbreviation for "trimmed mean".

Description

Only the start-tag is permitted. The content consists of three floating-point values representing the estimated or sample trimmed mean value for the component, the lower trimming fraction, and the upper trimming fraction. A fraction is used since all of the literature on trimmed means appears to use fractions. Since the normal practice is to trim symmetrically, the second and third values will typically be the same.

Format

The content of <tmean> is three floating-point values representing the trimmed mean, the lower trimming fraction, and the upper trimming fraction.

Example

<enerc> 354.2 FDS <tmean> 325 0.10 0.10 </enerc>

<smsz>

The <smsz> element is an immediate subsidiary of <*specific component*> elements. More specifically, it is a component of <*data description*> and follows the data value (and any <loctype> element, if present) associated with <*specific component*> or <*specific derived component*> elements. It is used to specify the effective sample size (sometimes called "N" or, confusingly, "sample population" or "pop") associated with the statistical estimates.

By "effective sample size" we mean the sample size after informal data cleaning and similar processes (which might be described with the <sclean/> element), that is, the sample size used in computing the other statistics reported. If outliers are removed from the data as part of a cleaning process and, e.g., a mean is reported that reflects the smaller data set, the <smsz> element should show the sample size with the outliers already removed and the value should be reported with <mean> or " <loctype> mean". However, if a fractional trimming process is used instead of subjective outlier elimination, <smsz> should show the sample size before trimming and <tmean> should be used to express the trimmed mean and the trimming fractions. "Smsz" may be thought of as an abbreviation for "sample size".

Description

Only the start-tag is permitted. The content consists of a single integer value representing the effective sample size for the statistical estimates. If the statistics represent "trimmed" values, the <smsz> element represents the sample size before trimming.

Format

The content of <smsz> is a single integer value representing the sample size.

Example

<procnt> 17.27 USDA 6.38 <serr> 0.5085 <smsz> 24 </procnt>

This would indicate that the mean and standard error values for protein were calculated from 24 samples.

<sdv>

The <sdv> element is an immediate subsidiary of <*specific component*> elements. More specifically, it is a component of <*data description*> and follows the data value (and any <loctype> element, if present) associated with <*specific component*> or <*specific derived component*> elements. It is used to specify the sample estimate of the population standard deviation value (i.e., with a denominator of N-1). "Sdv" may be thought of as an abbreviation for "standard deviation".

Description

Only the start-tag is permitted. The content consists of a single floating-point value representing the population standard deviation value for the component.

Format

The content of <sdv> is a single floating-point value representing the estimated population standard deviation.

Example

<CA> 9.4 <loctype> mean <sdv> 1.6 </CA>

The estimate of location is specifically identified as the mean in this example.

<serr>

The <serr> element is an immediate subsidiary of <*specific component*> elements. More specifically, it is a component of <*data description*> and follows the data value (and any <loctype> element, if present) associated with <*specific component*> or <*specific derived component*> elements. It is used to specify a standard error value for the food component. "Serr" may be thought of as an abbreviation for "standard error".

Description

Only the start-tag is permitted. ,The content consists of a single floating-point value representing the standard error value for the component.

Format

The content of <serr> is a single floating-point value representing the standard error.

Examples

<mg> 4 <serr> 0.5 <smsz> 17 </mg>
<thia> 0.025 <smsz> 15 <serr> 0.0032 </thia>

<jserr/>

The <jserr/> element is an immediate subsidiary of <*specific component*> elements. More specifically, it is a component of <*data description*> and follows the data value (and any <loctype> element, if present) associated with <*specific component*> or <*specific derived component*> elements. It is used to specify a non-parametric estimate of standard error based on data resampling [22, 6]. "Jserr" may be thought of as an abbreviation for "jackknife standard error".

Description

Both start-tag and end-tag are required. The content consists of a single floating-point value representing the standard error value, obtained by jackknifing, for the component and an optional <cmt/> element. The <cmt/> element should be used to describe special circumstances or assumptions associated with the jackknife procedure, e.g., grouping and the number of groups.

Format

The content of <jserr/> is a single floating-point value representing the standard error and an optional <cmt/> element.

Examples

<mg> 4 <jserr/> 0.5 </jserr/> <smsz> 17 </mg>

<thia> 0.025 <smsz> 5000 <jserr/> 0.0032 <cmt/> Jackknife on ten groups
</cmt/> </jserr/> </thia>

<cnfi>

The <cnfi> element is an immediate subsidiary of <*specific component*> elements. More specifically, it is a component of <*data description*> and follows the data value (and any <loctype> element, if present) associated with <*specific component*> or <*specific derived component*> elements. It is used to specify the confidence interval for the estimate of location of the food component.

The comprehensibility and usefulness of confidence intervals tends to be fairly low when calculated in conjunction with the fairly small sample sizes typical of food composition data. Also, while a confidence interval can, in principle, be computed for any statistic and may, as provided for here, be asymmetric, most readers will tend to construe it as a two-sided symmetric estimate for the mean. Other applications or situations should, if possible, be described with <cmt/> elements, or, preferably, other statistics and elements should be used.

"Cnfi" may be thought of as an abbreviation for "confidence interval".

Description

Only the start-tag is permitted. The content consists of three floating-point values representing respectively the lower confidence bound, the upper confidence bound, and the probability value, expressed as a fraction, for which the confidence interval is computed.

Format

The content of <cnfi> consists of three floating-point values representing the confidence interval and associated probability.

Example

<p> 16.0 <sdv> 2.2 <cnfi> 11.6 20.4 0.95 </p>

This food would have a phosphorus value between 11.6 and 20.4, with p=0.95.

<detect-lvl>

The <detect-lvl> element is an immediate component of <*specific component*> and <*specific derived component*> elements. It is used to identify the detection level of the instruments or method used to determine a particular value. It may be supplied for general information; its use is strongly recommended when a value is reported as "TR" (i.e., a trace), since a trace with one method might be a measurable value with another.

Description

Only the start-tag is permitted. The content consists of a single floating-point value in the same units as the estimate of location (for <*specific component*> elements) or the value (for <*specific derived component*> elements).

Format

The content of <detect-lvl> consists of a single floating-point value.

Example

<na> TR <detect-lvl> 0.005 </na>

The food contained a trace of sodium, but no amount below 0.005 mg could be detected and measured by the method and instruments in use.

<sclean/>

The <sclean/> element is an immediate subsidiary of <*specific component*> elements. More specifically, it is a component of <*data description*> and follows the data value (and any <loctype> element, if present) associated with <*specific component*> or <*specific derived component*> elements. It specifies the methods used to "clean" the data and their implications. "Sclean" may be thought of as an abbreviation for "sample cleaning".

Description

Both start-tag and end-tag are required. The content consists of elements only, there is no immediate data. The subsidiary elements include <cmt/>; other subsidiary elements will be defined in the future.

Format

The content of this element is an optional <cmt/> element and additional elements that will be defined as they are needed.

Example

```
<enerc> 354.2 FDS
<sclean/> <cmt/> outlier values eliminated by inspection </cmt/> </sclean/>
</enerc>
```

<edistr/>

The <edistr/> element is an immediate subsidiary of <*specific component*> elements. More specifically, it is a component of <*data description*> and follows the data value (and any <loctype> element, if present) associated with <*specific component*> or <*specific derived component*> elements. It is used to specify the empirical distribution for the value of the food component. "Edistr" may be thought of as an abbreviation for "empirical distribution".

Description

Both start-tag and end-tag are required. The content consists entirely of elements; there is no immediate data. Some elements are defined in this document; others will be added in the future. The defined subsidiary elements are <cmt/>, <bounds>, <mdbds>, <sum7>, and <pctpts>. Ordinary estimates of variation and estimates of the population distribution, e.g., the estimated population standard deviation or the confidence interval, are included as elements immediately subsidiary to the <*specific component*> or <*specific derived component*>*,* not as elements subsidiary to this element.

Format

The content of <edistr/> element consists of elements that describe the distribution of the data values. See the description of the subsidiary elements for examples.

<sdistr/>

The <sdistr/> element is an immediate subsidiary of <*specific component*> elements. More specifically, it is a component of <*data description*> and follows the data value (and any <loctype> element, if present) associated with <*specific component*> or <*specific derived component*> elements. It specifies detailed information about the subjective distribution for the value of the food component or, more specifically, beliefs about that distribution. "Sdistr" may be thought of as an abbreviation for "subjective distribution".

Description

Both start-tag and end-tag are required. The content consists entirely of elements; there is no immediate data. The defined subsidiary elements include <cmt/>; other elements will be specified in the future.

Format

The content of <sdistr/> consists of elements that describe subjective beliefs about the distribution of the data values. Until specific elements are defined, <cmt/> should be used with a free text description.

<bounds>

The <bounds> element is an immediate subsidiary of the <edistr/> element, used to list the minimum and maximum values encountered in the sample. The bounds are often erroneously called the "range", which is really the difference between the upper and lower bound.

Description

Only the start-tag is permitted. The content consists of two floating-point values representing the minimum and maximum values encountered in the sample data. These values can be misleading when reported for small samples and confused with actual minimum and maximum values in the population [24], so one of the distribution reports (described in the pages that follow) that provides more information and more obviously reflects the sample is to be preferred when adequate data are available.

Format

The content of <bounds> consists of two floating-point values representing, in order, the minimum value and the maximum value of the sample data.

Example

<NA> 50 <loctype> mean <edistr/> <bounds> 35 90 </edistr/> </NA>

<mdbds>

The <mdbds> element is an immediate subsidiary of the <edistr/> element, used to list the median, hinges, and bounds of the distribution of the data.

Description

Only the start-tag is permitted. The content consists of five floating-point values representing the bounds, hinges (robust estimates of the quartiles), and median of the sample data [34]. Since the data in <mbds> are a subset of those represented by <sum7>, <mdbs> should be omitted if there are sufficient data to include <sum7> .

Format

The content of <mdbds> consists of five floating-point values representing, in order, the minimum value, the lower hinge, the median, the upper hinge, and the maximum value of the sample data. These values, also known as a "five-number summary" can be used to summarize the distribution of the sample data.

Example

<mdbds> 72 79 86.5 90 92

<sum7>

The <sum7> element is an immediate subsidiary of the <edistr/> element, used to list the median, fences, and bounds of the distribution of the data. "Sum7" may be thought of as an abbreviation for "seven-number summary".

Description

Only the start-tag is permitted. The content consists of seven floating-point values representing the bounds, fences, hinges, and median of the sample data [34].

Format

The content of <sum7> consists of seven floating-point values representing, in order, the minimum value, the lower fence, the lower hinge, the median, the upper hinge, the upper fence, and the maximum value of the sample data. These values, known as a "seven-number summary" can be used to summarize the distribution of the sample data.

Example

<sum7> 72 78 79 86.5 90 90.2 92

<pctpts>

The <pctpts> element is an immediate subsidiary of the <edistr/> element. It specifies the percentage points of the actual distribution of the sample data for some food component. For a given percentage point, the data value provided is such that the percentage of the data shown is smaller than the data value. See <locpctl> for a discussion of a slightly related location statistic. "Pctpts" may be thought of as an abbreviation for "percentage points".

Description

Only the start-tag is permitted. The content consists of ordered pairs of values, where the first member of each pair represents the percentage value, expressed as a fraction, and the second one represents the data value for that population percentile. The statistical literature is not consistent about whether percentage points should be expressed as percentages or fractions. We have chosen fractions for this element in the hope that they will make table or data base checking slightly easier. By convention, the 0 and 100% (1.0) percentage points are rarely reported, but, if they are, they are respectively the minimum and maximum values that appear in the sample.

Format

The content of <pctpts> is a sequence of floating-point values representing the percentage value (as a fraction) and the data value at that percentile. In other words, the format is

<pctpts> *percentile1 datavalue 1 percentile2 datavalue2 ...*

Example

<pctpts> .60 .38 .80 1.25 .95 2.44

Part III: Processing data and interchange files

7. Registering elements

INTRODUCTION

All systems which attempt to facilitate communication between different parties must develop a set of rules to which those parties must adhere. In the case of telephone or television transmission, these rules are largely invisible to the average user. The physical devices themselves are built according to agreed-upon standards and usually perform their functions without our having to think about them.

In the case of intra- and international exchange of data, where system independence is prerequisite, rules must also exist which allow all parties to participate in data interchange as efficiently, and with as few errors and misunderstandings, as possible. A goal of INFOODS is to create a mechanism to make interchange of food composition data as invisible as telephone or television transmission mechanisms. That goal is not yet possible, both because of differing standards about the data values themselves- analogous to two people trying to talk without speaking or understanding each other's languages-and because some identification issues, such as "When are two foods 'the same'?", require problem-specific scientific determination.

JUSTIFICATION FOR REGISTRATION

The interchange system is, as discussed in earlier chapters, a "tagged architecture" in which the meaning of each data value is specified by the generic identifier-more specifically, the structure of the element-with which it is associated. This obviously implies that the accurate matching of tagging structure to data is critical to the interpretation of those data: if a value for fat were somehow identified as a value for vitamin A, the distortion of values might be serious. Fortunately, a properly organized tagged architecture is less prone to misidentification of values than, say, an approach that depends on the order in which the data values appear. But permitting data to be moved among systems is not the only goal of the interchange system. Other goals include permitting those data to be exported and imported with very high accuracy and no loss of information and also being able to adapt to improving knowledge about nutrients and their analysis over time. Meeting those goals depends, to a significant degree, on agreement about generic identifiers and element structures between those who develop or send data and those who receive them. If an element appears in an imported interchange file, the receiver must be able to determine, efficiently and exactly, what that element, and all of its components, means.

Consequently, it is necessary to define the generic identifiers and elements, and the meaning of the data values, unambiguously and very precisely. That the initial listing of food components and associated tagnames [17] took four review cycles and two years to complete is an indication of the difficulty of the process. Perhaps more indicative: more information and discussion of subtle differences between variants on what is usually thought of as the same nutrient were needed (and added) during each of these cycles.

The initial list represented by these two documents cannot, obviously, be comprehensive for all time. New food components of interest will be identified, and improved methods, yielding different values, will be developed for food components now commonly reported. In addition, new elements will need to be defined for additional statistical and sample description and for non-nutritive components of foods such as additives and contaminants. Consequently, an integral part of the interchange system must be a mechanism for defining those new element structures and, in some cases, modifications to already-defined elements. That mechanism, following the broad concepts of an ISO model, involves the use of a "registration authority" for each type of element that can be defined. The INFOODS secretariat initially holds all registration responsibility and, as in the case of the food component tagnames, has gone beyond simple registration to a leadership role in defining the elements. It is expected that, in the future, other organizations will assume some element registration responsibilities, and that submissions of new element definitions will come from data user or producer organizations or from regional groups.

In any case, however elements originate, it is critical that they be precisely defined and registered, and the definitions made available, before their use in interchange is attempted. Since a major goal of the interchange system approach is to eliminate the need for the prospective receiver of a data file to have separate conversion programs for each organization from which data might be received, the receiver and the programs that support conversion from interchange format to the receiver's local formats must be able to accurately anticipate the structure and organization of any valid incoming interchange-format file. This goal is consistent with being able to expand the list of permissible elements over time only if there are dear rules about which types of elements can be ignored if not recognized; those rules are described in this chapter and elsewhere in this book.

This chapter defines the activities and responsibilities of the registration authorities, and specifies the conventions and requirements for proposing new elements and having them adopted. In principle, the structure of the interchange system can be extended in ways not specified in this chapter, such as by the addition of new "structural" elements. But doing so is not in the same category as the registration of a new data element: new structural elements can alter the rules about what can safely be ignored and would require fairly general consensus to adopt. Consequently, while the interchange model anticipates the possibility of such changes, no specific mechanism for making them is included here.

THE REGISTRATION AUTHORITY

Registration is a secretariat function, with little technical responsibility other than ensuring that materials required as part of a registration request are in order, complete, and consistent with previous registrations. There must also be a procedure for disseminating what has been registered. In the case of food composition data, only elements are registered. Those elements may identify new food components, or new modes of statistical description, or new sampling strategies, or new components of food descriptions, to name a few. In some cases, discussed below, a new submission for registration may also modify an existing registered definition, such as adding to a list of keywords that describe methods or conversion factors.

POLICY AND PROCEDURES FOR ACCEPTANCE

The registration authority must ensure that all elements are unique and that descriptions for each are complete and well-defined. The registration authority must also verify that relevant reference documentation is available. Neither definitions nor the actual elements themselves may be duplicated: one may neither have two definitions for the same name nor have two names for the same definition.

Details of the requirements and format of an application to register an element follow. Once the registration proposal has been submitted, the registration authority will evaluate the application. As mentioned above, applications to register elements are evaluated solely on uniqueness and completeness. The registration authority is not normally expected to evaluate the relative merits of the analytic methods used or to apply other qualitative criteria.

When an application is accepted, the appropriate registration authority assigns both a generic identifier for the element and whatever registration identification is required to catalogue and retrieve element definitions and cross-references over time.

REGISTRATION REQUIREMENTS AND FORMAT

In general, a registration proposal must, in addition to identifying the applicant and any review procedures already applied to the proposal, completely define the proposed element or modification, its context, area of application, and relationship to other elements that might be confused with it. The subsections that follow list this information in considerable detail, and may safely be omitted by the casual reader.

Required Information

The following items are required and must be included in a proposal to register a new element. They are described in more detail below.

- The name and address of the applicant.
- The context in which the proposed element is used and the category to which it belongs, e.g., to what elements it is subsidiary.
- The definition and justification for the element.
- The proposed generic identifier.
- A description of the element's content, with any information relevant to its accurate interpretation. In particular, for elements that identify new food components (i.e., elements subsidiary to <food> and <comp> or to <food> and <drvd-comp>), detailed information on common and default units of measure, synonyms, data tables, analytic methods, and distinguishing characteristics. Information comparable to that in the existing lists of elements in this area [17] should be provided.
- A list of relevant subsidiary elements that can be used with the proposed element in the interchange hierarchy, with complete syntax or syntax references.
- Any cross-references.
- Example(s) of the proposed element.

Each of these items must be specified precisely and completely, according to the definitions and guidelines that follow.

Context within the Interchange System

Every defined element of the interchange system is subsidiary to some other element except for the root element, <infoods 85>. The small number of structural elements identified in this document-<header>, <dflt>, and the immediate subelements of <food>: <classif> <fddflt>,

<comp>, and <drvd-comp>- provide a context for ail other elements, including those not yet defined. Elements that are not directly subsidiary to the structural elements, but that lie further "down" in the structure, also draw context from the elements to which they are directly subsidiary. A proposal for a new element must include the context or contexts in which that element may appear.

Definition and Justification

This portion of the proposal must provide a definition or description of the proposed element. This description may be informal, but must be sufficient to permit someone to distinguish between one element and another. For example, for primary nutrient elements, this section must include the name of the nutrient and, if appropriate, the analytic method that distinguishes it. If there is already an element defined for the same or a similar purpose, justification must be provided as to why the existing element is not sufficient and the differences between the older and newer one(s) defined very clearly. For obvious reasons, the registration proposal for a completely new food component will be less complex than a proposal to define a new element where similar elements already exist.

Proposed Generic Identifiers

Generic identifiers provide the identification of an element, and are typically the names by which the element is known, indexed, and referenced. For the sake of readability and use, they should ideally be from three to seven characters in length, and, if possible, in pronounceable or nearly-pronounceable strings (this criterion is often not practical). When possible, names that have mnemonic significance in some language are preferred to those that are completely arbitrary. Most of the initial generic identifiers were derived from Latin, English, or chemistry.

In order to reduce the risk of undetected transcription errors, generic identifiers shorter than three characters are discouraged, except under special circumstances (see the introduction to the reference sections), as when the generic identifiers are names or abbreviations in nearly universal use. On the other hand, generic identifiers over eight characters are discouraged in order to minimize the size of, and amount of processing required for, an interchange file. Just as short names will be permitted when there is strong reason for doing so, longer names may be permitted when more important principles apply. For example, the uniform system used to assign the initial food component tagnames for fatty acids led to several generic identifiers that were more than eight characters long. In this case, consistency was considered more important than brevity.

The character strings (names) used for generic identifiers must start with a simple alphabetic character followed by simple alphanumeric characters and, under restricted circumstances, hyphens. The "simple" alphabetic characters are selected on a common denominator basis as the alphabetic characters common to Latin-based alphabets, without diacritical marks, special symbols, or characters designated as "national use" in the various international character coding standards. Simple alphanumeric characters are the simple alphabetic characters plus the digits. Obviously, generic identifiers may not contain embedded blanks or other "whitespace" or non-graphic characters. A more precise description of the characters permitted, and the associated rules for using them, appears in Chapter 3. In spite of the comments above about mnemonic significance and standard abbreviations, the generic identifiers of the interchange system are ultimately arbitrary character strings: programs may not assume that similar-looking generic identifiers are related, and people should be discouraged from making that assumption.

While the registration authority must accept a complete, consistent, and new definition, the proposed generic identifier is just a recommendation: the registration authority will make final decisions on matters of taste in generic identifier assignment.

Description of the Content

The content entry specifies the values, and characteristics of those values, for the proposed element. This includes, as discussed above, a description of the meaning of the element when optional content components-typically keywords or subsidiary elements- are omitted. The description of the content will frequently include references to the subsidiary elements and cross-references, provided above, for the sake of completeness. Where there are existing applicable international standards-such as ISO, Codex Alimentarius, IUPAC, or AOAC definitions, standards, or units-they are preferred to other alternatives and should be used and referenced.

For nutrient elements, the description of the content should include the number of values that are required and permitted (if different) and how they are to be interpreted. "Interpretation" information includes the units that a numerical value represents (in units/unit form, e.g., "grams per hundred grams edible portion"). If some values are optional, this section must indicate what their omission means.

For values that are expected to be numeric, the plausible range should be given when that is useful. If this range can be modified by subsidiary elements, that fact should also be indicated along with what variations are possible. However, modification of values by subsidiary elements is not desirable. Ideally, a receiver who ignores subsidiary elements below a certain level should not encounter serious problems. With the exception of <unit/>, this principle is followed in all of the initial sets of element definitions, and the implications of <unit/> are restricted, as discussed below.

"Units" which express a scale, e.g., "expressed as an integer, to be divided by 1000", are not permitted since they put an excessive premium on external knowledge. Notation for the values themselves that uses an explicit scale will be used instead (e.g., "5.2E-3"). This should not be taken as precluding the use of common SI multiplier units, such as milligrams: those are explicitly permitted. It is strongly preferred that the default unit for a particular element should be the one in most common use and that is scientifically most acceptable to permit the <unit/> to be omitted in most cases.

For classification and other descriptive elements, the description of the content must include either a complete list of the values that may appear, a reference to where such a list may be found, or a very specific "generating rule" that can be used to determine what may or may not appear in the value. A "generating rule", as the term is used here, is a rule about what values are permitted and what they mean without listing the values. For example, "the quality value is a positive integer less than six" is a generating rule, while "the quality value must be one of 1, 2, 3, 4, or 5" is a complete list of values. Neither is a complete description of a content, since the meaning and interpretation of the values is not given.

If a list or generating rule is incorporated by reference, the reference must be specific and must refer to materials that are readily accessible to the scientific community. For example, this reference form is acceptable: "the Australian Food Composition Tables, Government of Australia, 2 January 1903" since it is a specific reference to readily available material. Conversely, this reference form is unacceptable: "the current version of the 'Factored Food Vocabulary'" since "the current version" is not wet/-defined. To be acceptable, a specific version and source for obtaining it would need to be provided. Even "the version in use on 1 June 1988" is not sufficiently specific, as there is no reference to a document that is readily accessible to the scientific community.

While having a document filed with the registration authority is not sufficient to make it "readily accessible to the scientific community", deposit of referenced materials is also required unless they are very widely known and accessible.

Keyword Content Values

In specific cases, contents are composed of keywords or keywords and values. A keyword is a member of a controlled list of possible values (usually best thought of as names) for some item of information. The list is always restricted, and is often an alternative to the use of long descriptions (e.g., the keyword used for the name of a language) or a complex list of conversion factors or similar values (e.g., the keyword "CODEX" used with the <enerc> element to indicate the Codex Alimentarius-recommended energy conversion factors). Keywords will usually be registered and maintained by the appropriate registration authority, just as generic identifiers are, but a registration proposal may incorporate an international standard by reference. For example, there is an ISO standard for the representation of names of languages [39] which forms the list of keywords for the <lang> generic identifier.

Example

<lang> AR
<unit/> KCAL </unit>

References for registered keywords are maintained by the registration authority with the keyword registration materials, in lieu of a specific list of keyword values.

List of Relevant Subsidiary Elements

In many cases, an element will permit or require additional elements as part of its content. Typical subsidiary elements might identify units of measure different from the default for the food component, analytic methods that do not alter the expected values for a nutrient (different analytic methods that produce different expected values for the same nutrient call for different elements, pairing the nutrient with the method), statistical or sample description of the values presented, or other qualifying or descriptive information.

The definition for a subsidiary element is, in principle, identical to the definition for an element. This section should identify subsidiary elements by reference to their definitions, and provide information as to whether they are requited or optional in this particular context. Any constraints on subsidiary element values should also be explained, using the general style discussed under cross referencing above. New subsidiary elements themselves may be defined either as part of the registration proposal for the parent element or, especially if their use in other contexts is anticipated, in separate, but concurrently submitted, registration proposals.

Required subsidiary elements are discouraged in order to make simple processors easier to construct, but necessary exceptions may arise and should be justified. For example, the <enerc> element requires, in addition to the nutrient value, either a keyword specifying a calculation method or subelements that list the specific conversion factors used. Without one or the other, the energy value cannot be adequately identified and interpreted, and the use of that particular element is not permitted (<enerc> must be used instead). When required subsidiary elements are needed, as in this case, a justification similar to this example should be given, preferably with a discussion of why a series of separate generic identifiers and associated elements are not a preferable solution.

The reason for avoiding required subsidiary elements when possible is to avoid processing complexity, especially for users who are seeking only a particular type of information with relatively modest software. Elements should be defined in a way that is consistent with the most accepted or common use, to minimize the amount of qualification that is required except in the more obscure or unusual cases. On the other hand, since there may not be general agreement about the most accepted use, the meaning of the element without any optional qualifying information should be clearly specified. Optional qualifying elements also tend to increase the machine size and code complexity needed to deal with the interchange system and should be avoided, when feasible, on those grounds as well.

Cross-references

In those cases in which a proposed element is closely related to one or more other elements (e.g., a new method, producing different expected values, for a nutrient for which elements are already registered), the registration proposal must identify the earlier registrations and what the relationship is between the existing elements and the proposed element. The registration authority will maintain and update this section in the permanent reference copy of the element definitions. It should be noted that registration cross-references are intended for understanding, interpreting, and maintaining the list of elements and the interchange system in general. Although it is not its primary purpose, the information may also be of use when sites or regions develop thesauri to expand or automatically generate data base searches.

In particular, if cross referencing is needed, the cross-references must not become convoluted. For example, they must refer to an original element definition, not to other cross-references. One should avoid "refers to element '<xxx>' as used subsidiary to element '<yyy>' with the modifications and constraints of '<zzz>'" because this type of reference rapidly leads to confusion and ambiguity.

In addition, while cross-references may constrain or qualify the values or definitions of an original element, they may not expand those values or definitions. For example, one might say "refers to element '<qqq>', except that, in this context, the value 'pounds per cubic meter' accepted in the general definition of that element, is not permitted" because it refers to another element but constrains its use. However, one may not say "refers to element '<m>', except that, in this context, the value 'pounds per cubic meter' accepted in the general definition of that element, is not permitted and instead substitute the value 'pounds per cubic inch'" because it both restricts and expands upon the original element definition, leading to some convolution of definitions.

A more adequate definition of the term "convolution", and the purpose of these restrictions, is to avoid definitions which a person (or computer) must construct dynamically by referencing several different pieces of text. Dynamically constructed definitions are confusing and annoying for the reader and, especially in extensible systems such as the interchange definition, are error-prone and subject to ambiguities.

REGISTERING A KEYWORD TO BE ADDED TO AN EXISTING LIST

Certain keyword lists may be expanded. An application to register a new keyword, that is, to extend an existing keyword list, must include the proposed word, its meaning, and the context (element) in which the keyword will appear. It is, of course, not possible to expand a keyword list established by reference to an international standard except by revising that standard. The reference sections identify the existing elements for which keyword lists may be extended. New registrations must clearly present the model, if any, for expanding element definitions with new keywords or other syntax.

TRACE AND MISSING VALUES

In the interchange system, "missing" data is never represented by a value, but by the omission of something-a value or an element. All situations in which values may be "missing" and what that situation means must be clearly identified. See Chapter 3 and Stewart's comments [24] for discussions of the representation of missing values in interchange files and food composition data more generally.

CONCLUSION

See the appendices for application forms for registering elements.

8. Conversion of data to interchange format

INTRODUCTION

Displaying data in interchange format involves printing out each data element in the proper order, and inserting tags and occasional whitespace between them. Additional whitespace (including line and record breaks) can be inserted to keep the line or record length below a system-mandated maximum or to make a listing of the interchange file easier for people to read. This chapter will discuss mechanisms for producing interchange files from "paper copy" and machine-readable data, the latter either directly maintained or accessed through a data base management system.

MANUAL INPUT OF DATA

If food data to be placed in an interchange format file are not in machine-readable form, and are not intended to be made so except for creating the file, then the appropriate way to create an interchange file is to use a word processor and manually enter the data for subsequent conversion into interchange format. If the data are irregular (e.g., different components represented for each food) or include only a few entries, it may be appropriate to enter the interchange tags themselves at the same time.

If the data consist of entries for the same set of food components for each of several or many foods, then it may be desirable to insert the necessary tags semi-automatically. This could, for example, be done within some editors by creating a "macro" that will insert all of the tags for a single food's entries. For example if the editor being used has the ability to capture keystrokes, the sequence of operations (assuming the data values are entered one per line) would be to align the editor's current position at the first datum for the first food, and then capture:

- Type in the opening "<food>" and the first datum's start-tag(s) (perhaps" <classif> <ifri> ").
- Position to the end of the line. (Do this by using the editor's position-to-end-of-line command, not by moving over one character at a time, so the macro will always go to the end of the line no matter how long it is.)
- Type in the closing end-tag(s) appropriate for this item. (This may or may not include higher-level tags such as "</classif>" or, later on, "</comp>".)
- Position to the beginning of the next line. Type in this item's start-tag(s). Position to the end of the line. Type in the end-tag(s).
- Repeat until all of the items for this food are tagged.
- Type in the closing "</food>".

Turn off the keystroke-capturing. Position to the beginning of the next food and trigger a rerun of the captured keystrokes (i.e., execute the macro). This will automatically insert all of the tags for this food. Repeat until all foods are tagged. Then don't forget to manually insert the <infoods 85> tags and the initial <header> and <dflt> elements.

This same addition of tags could be accomplished by a small specially written program. For example, the following subroutine (written in the BASIC language for illustration) will add the necessary tags. (Note that the subroutine must be specially written for a specific sequence of data values for each food.)

```
100 input #1, datavalue$
write #2, "<food><classif><ifri>"; datavalues;" </ifri>"
input #1 datavalues
write #2 " <bvname>"; datavalues;" </bvname></classif>"
input #1, datavalue$
write #2, " <comp>"
```

and appropriate tags surounding "datavalue$"

repeat the input/write pair as needed

```
write #2, "</food>"
return
```

This subroutine accomplishes much the same thing as the editor macro just described. The only difference is that instead of modifying each line, the datum-only input line is copied onto the output file surrounded by the appropriate tags. The complete program must ensure that the first datum of each food is found, and will have to loop, recalling the subroutine until the last food's data is read. This will presumably be detected by end-of-file on the input file of raw data. If an editor is available, it is probably still easier to place the <header> and <dflt> elements et the beginning of the fife using the editor rather than building them into the program-unless the same program will be used more than just a few times.

Of course, a program could also be written that would prompt for the data to be provided as it is being tagged and written out. This might be the best means of making the file if an editor program is not available; it depends upon the availability of programmer time and skills to build such a program.

MACHINE-READABLE SOURCE DATA

The first step in dealing with non-machine-readable data is to key or scan it into the machine, and then process it into interchange format. If the source data is maintained in machinereadable format, there are two possibilities (other than the data base management system environments discussed in the next section):

On the one hand, the data might be stored as numerals and other character strings; in this case the data can be processed just as described for newly keyed-in non-machine-readable data. Such data are likely, however, to need sorting to get them into the order required by the interchange format: <ifri> and <bvname> before any food component data; <comp> data before <drvd-comp> data; etc. If all of the data for a single food can be read in before any are written out, this should not be difficult since the order of data in the input format and the interchange format is known in both cases.

It is also possible that the component data does not have the same components for every food. This is the case with the USDA data base [35]. In such cases, each data set for a component must necessarily include information identifying the component. Most ordinary text editors do not have the capability to look up generic identifiers for the various components; a specially written program will probably be needed. The program must determine the generic identifier, delete the identifying information, reorder the remaining data and tag them as appropriate, and then surround the result with the correctly generated start-tag/end-tag pair.

On the other hand, the numerical data might be stored in a "binary" representation, i.e., an internal numerical representation other than numerals made up of digits drawn from ISO 646 [40]. Few editors are suitable for this type of data; a special program will probably be needed. Existing subroutines that can read-in such data for one purpose or another should be identified; they are good starting points for creating a local-format-to-SGML conversion program.

Once the data for a given food's components are read, the values are written with appropriate bracketing tags. For example, the following BASIC subroutine will write out a <food> element for one food with a single local name and two nutrients:

```
100 WRITE #2, "<food>"
WRITE #2, " <classif>"
WRITE #2, " <ifri> ";IFRIS;" </ifri>"
WRITE #2, " <bvname> ";LOCALNAMES;" </bvname>"
WRITE #2, " </classif>"
WRITE #2, " <comp>"
WRITE #2, " <CA> "; CAS;" </CA>"
WRITE #2, " </comp>"
WRITE #2, "</food>"
WRITE #2, " <FE> ", FES;" </FE>"
RETURN
```

If, for example, the data values for calcium and iron were loaded as numbers rather than numerals (i.e., character strings), this subroutine listing would have "CA$" and "FE$" replaced respectively by "CA" and "FE".

If, in accordance with that example, the values for "LOCALNAME$", "CA$", and "FE$" are set respectively to "Banana", "5.7", and "63", and IFRI$ is set appropriately before the subroutine is called, then the following will appear in the output file:

```
<food>
<classif>
<ifri> an appropriate IFRI </ifri>
<bvname> Banana </bvname>
</classif>
<comp>
<CA> 5.7 </CA>
<FE> 63 </FE> </comp>
</food>
```

SOURCE DATA IN A DATA BASE MANAGEMENT SYSTEM

If a food composition data base is maintained within a data base management system, it will usually be most convenient to use that system's report writer to produce results in interchange format. The techniques will be similar to those just illustrated, but the "programming" language used will be that of the system's report writer. The examples below omit the <ifri> element as well as much of the identification and classification information that might be present in a production data base.

Example: The Janus data base management component Of the Consistent System.

A program called "Janus" [16, 20, 29], designed as a data organizing and management front end for the Consistent System [5], is fairly representative of the state of the art in data systems specifically designed for the management and handling of statistical data. It was included in the experiments run at the INFOODS Secretariat because it contains a collection of special operators and operations that make working with nutrient composition data quite simple. Although the system is essentially relational, the Janus design both predates SQL by some years and supports operations that do not mesh well with the relational model [18].

The Janus Command Sequence

Janus uses a non-procedural command language for all operations; while there is a host language interface, it is rarely used and should never be necessary (see [16]). Its formatted output operations occur through a command called "display", and the [single] command used to produce the output shown below is as follows:

```
display cntel ("<FOOD> <CLASSIF> <FDA-FFV-8807>"), FFV, cntel ("</FDA-FFV-8807>
<USDA-NDB>"), ndb,
cntel ("</USDA-NDB> <EUROCODE2>"), eurocode,
cntel ("</EUROCODE2> <USDA-NAME>"), usda name,
cntel ("</USDA-NAME> </CLASSIF>") cntel ("<COMP> <ASH>"), ash,
cntel ("</ASH>"), cntel ("<CHOCDF>"), carbo, cntel ("</CHOCDF>"),
cntel ("<ENERC>"), energy268, cntel ("USDA </ENERC>"),
cntel ("<FIBC>"), fiber, cntel ("</FIBC>"),
cntel ("<PROCNT>"), protein cntel ("JONES </PROCNT>"),
cntel ("<FAT>"), total lipid cntel ("</FAT>"),
cntel ("<WATER>"), water,
cntel ("</WATER> </COMP> </FOOD>"),
in extract foods with no blocking, no enn, lnl=600
save in Mlf ">udd>INFOODS>jck>extract4.list";
```

In this command, the names "ndb", "eurocode", "usda name", and so forth are names of fields (which Janus calls "attributes", reflecting early relational data base management system terminology). The function "cntel" is used to force its value to be displayed along with the field (attribute) values, rather than being separated as if they were summary fields. The qualifications following "with" are specifications about the display output format, specifying that selected Janus display defaults are to be disabled or overridden. See the Janus Reference Manual [20] if additional information is needed.

The data used in this example are extracted from the USDA Standard Reference Database [35] with factored food vocabulary codings supplied by FDA prior to the workshop discussed in reference 13. Since the records being produced do not contain <ifri> elements, they are not complete interchange records.

The Janus Output

The following output has been reformatted to fit better on a page. The command specification given above produces a single "line" per food record. Additional syntax could have been supplied to the Janus display command to force the indention shown, but the formatting is not required for the interchange format and would make the syntax example considerably more obscure.

```
<FOOD>
<CLASSIF>
<FDA-FFV-8807> E134 A185 C245 B1201 P24 N01 M001 K03 J001 H101 F14 </FDA-
FFV8807>
<USDA-NOB> 1014 </USDA-NOB>
<EUROCODE2> 1.54 </EUROCODE2>
<USDA-NAME> CHEESE: NATURAL, COTTAGE, UNCREAMED, DRY, LARGE OR
SMALL CURD
</USDA-NAME>
</CLASSIF>
<COMP>
<ASH> 0.690 </ASH>
<CHOCDF> 1.850 </CHOCDF>
<ENERC> 354.027 USDA </ENERC>
<FIBC> 0.000 </FIBC>
<PROCNT> 17.270 JONES </PROCNT>
<FAT> 0.420 </FAT>
<WATER> 79.770 </WATER>
</COMP>
</FOOD>

<FOOD>
<CLASSIF>
<FDA-FFV-8807> C245 H247 A281 J001 F14 B1201 K03 M001 E125 H107 P24 N01
</FDA-FFV-8807>
<USDA-NOB> 1035 </USDA-NOB>
<EUROCODE2> 1.51 </EUROCODE2>
<USDA-NAME> CHEESE: NATURAL, PROVOLONE </USDA-NAME>
</CLASSIF>
<COMP>
<ASH> 4.710 </ASH>
<CHOCDF> 2.140 </CHOCDF>
<ENERC> 1471.019 USDA </ENERC>
<FIBC> 0.000 </FIBC>
<PROCNT> 25.580 JONES </PROCNT>
<FAT> 26.620 </FAT>
<WATER> 40.950 </WATER>
</COMP>
</FOOD>

<FOOD>
<CLASSIF>
<FDA-FFV-8807> C245 A111 F14 H107 J001 K03 M001 N01 P24 B1201 E001
</FDA-FFV-8807>
<USDA-NOB> 1046 </USDA-NDB>
<EUROCODE2> 1.56 </EUROCODE2>
<USDA-NAME> CHEESE FOOD: PASTEURIZED PROCESSED, AMERICAN, W/O Dl
```

```
NA PHOS </USDA-NAME>
</CLASSIF>
<COMP>
<ASH> 5.350 </ASH>
<CHOCDF> 7.290 </CHOCDF>
<ENERC> 1373.437 USDA </ENERC>
<FIBC> 0.000 </FIBC>
<PROCNT> 19.610 JONES </PROCNT>
<FAT> 24.600 </FAT>
<WATER> 43.150 </WATER>
</COMP> </FOOD>

<FOOD>
<CLASSIF>
<FDA-FFV-8807> J135 M001 P24 N01 A148 B1201 C113 E123 F18 H001 K03
</FDA-FFV-8807>
<USDA-NOB> 1049 </USDA-NOB>
<EUROCODE2> 1.01 </EUROCODE2>
<USDA-NAME> CREAM: FLUID, HALF & HALF, CREAM AND MILK </USDA-
NAME>
</CLASSIF>
<COMP>
<ASH> 0.670 </ASH>
<CHOCDF> 4.300 </CHOCDF>
<ENERC> 545.578 USDA </ENERC>
<FIBC> 0.000 </FIBC>
<PROCNT> 2.960 JONES </PROCNT>
<FAT> 11.500 </FAT>
<WATER> 80.570 </WATER>
</COMP>
</FOOD>

<FOOD>
<CLASSIF>
<FDA-FFV-8807> M001 N01 P24 J135 F18 H221 C235 B1201 A182 K03 E123 H208
</FDA-FFV-8807>
<USDA-NOB> 1059 </USDA-NOB>
<EUROCODE2> 1.03 </EUROCODE2>
<USDA-NAME> MILK: FILLED, FLUID, W/BLEND OF HYDR VEGETABLE OILS
</USDA-NAME>
</CLASSIF>
<COMP>
<ASH> 0.800 </ASH>
<CHOCDF> 4.740 </CHOCDF>
<ENERC> 264.280 USDA </ENERC>
<FIBC> 0.000 </FIBC>
<PROCNT> 3.330 JONES </PROCNT>
<FAT> 3.460 </FAT>
<WATER> 87.670 </WATER>
```

```
</COMP>
</FOOD>

<FOOD>
<CLASSIF>
<FDA-FFV-8807> P24 N01 M001 K03 J133 H161 F14 B1201 E106 A148 C235
</FDA-FFV-8807>
<USDA-NOB> 1091 </USDA-NOB>
<EUROCODE2> 1.3 </EUROCODE2>
<USDA-NAME> MILK: COW, DRY, SKIM, NON-FAT SOLIDS, REGULAR,
WO/ADDED VIT A </USDA-NAME>
</CLASSIF>
<COMP>
<ASH> 7.930 </ASH>
<CHOCDF> 51.980 </CHOCDF>
<ENERC> 1516.368 USDA </ENERC>
<FIBC> 0.000 </FIBC>
<PROCNT> 36.160 JONES </PROCNT>
<FAT> 0.770 </FAT>
<WATER> 3.160 </WATER>
</COMP> </FOOD>
```

Example: The Oracle commercial data base management system

Oracle is a fairly typical example of a data base management system. It represents technology which is the subject of US (ANSI) and international (ISO) standards and which has already become quite common and, in some fields, dominant. It uses the operations of the SQL language, supplemented by a specialized report generator. The commands below show the report formatting and definition for Oracle using SQL [26], and are similar to those that would be required for any similar system.

The Oracle Commands

```
select '<FOOF> <CLASSIF> <FDA-FFV-807> '||FFV||' <FDA-FFV-8807>'
'<USDA-NDB> '||nbd||' </USDA-NDB>'
<'EUROCODE2> '||eurocode2||' </EUROCODE2>'
'<USDA-NAME> '||usda name|| '</USDA-NAME> </CLASSIF>'
'<COMP> <ASH> '||ash||' </ASH>
'<CHOCDF>' ||chocdf|| '<CHOCDF>'
'<ENERC>' ||enerc||' USDA </ENERC> <FIBC> '||fibc||' </FIBC>'
'<PROCNT> ||procnt||' JONES </PROCNT> <WATER> '||water
'</WATER> </COMP> </FOOD>' from extract_foods;
```

In this command, the names "ffv", "eurocode2", "usda name" and so forth are names of fields (which Oracle calls "columns"). The "||" characters are concatenation symbols, which permit the text and values to be printed consecutively.

The Oracle Output

As in the Janus example above, the following output has been reformatted to fit better on a page. The SQL statement in the Oracle example above produces a single "line" per food record. Oracle provides report formatting capability [21] which could force the indention shown below. As in the Janus example, formatting is not required for the interchange format and, in the Oracle example, would make the syntax example many pages longer and considerably more complex and obscure. The report formatting in Oracle is also not part of the standard SQL language [38, 54].

```
<FOOD>
<CLASSIF>
<FDA-FFV-8807> E134 A185 C245 B1201 P24 N01 M001 K03 J001 H101 F14
</FDA-FFV-8807>
<USDA-NOB> 1014 </USDA-NOB>
<EUROCODE2> 1.54 </EUROCODE2>
<USDA-NAME> CHEESE: NATURAL, COTTAGE, UNCREAMED, DRY, LARGE OR
SMALL CURD </USDA NAME>
</CLASSIF>
<COMP>
<ASH> 0.690 </ASH>
<CHOCDF> 1.85 </CHOCDF>
<ENERC> 354.027 </ENERC>
<FIBC> 0.000 </FIBC>
<PROCNT> 17.270 </PROCNT>
<WATER> 79.770 </WATER>
</COMP>
</FOOD>

<FOOD>
<CLASSIF>
<FDA-FFV-8807> C245 H247 A281 J001 F14 B1201 K03 M001 E125 H107 P24 N01
</FDA-FFV-8807>
<USDA-NOB> 1035 </USDA-NOB>
<EUROCODE2> 1.51 </EUROCODE2>
<USDA-NAME> CHEESE: NATURAL, PROVOLONE </USDA-NAME>
</CLASSIF>
<COMP>
<ASH> 4.710 </ASH>
<CHOCDF> 2.14 </CHOCDF>
<ENERC> 1471.019 </ENERC>
<FIBC> 0.000 </FIBC>
<PROCNT> 25.580 </PROCNT>
<WATER> 40.950 </WATER>
</COMP>
</FOOD>

<FOOD>
<CLASSIF>
<FDA-FFV-8807> C245 A111 F14 H107 J001 K03 M001 N01 P24 B1201 E001
```

</FDA-FFV-8807>
<USDA-NOB> 1046 <
/USDA-NOB>
<EUROCODE2> 1.56 </EUROCODE2>
<USDA-NAME> CHEESE FOOD: PASTEURIZED PROCESSED, AMERICAN, W/O DI
NA PHOS
</USDA-NAME>
</CLASSIF>
<COMP>
<ASH> 5.350 </ASH>
<CHOCDF> 7.29 </CHOCDF>
<ENERC> 1373.437 </ENERC>
<FIBC> 0.000 </FIBC>
<PROCNT> 19.610 </PROCNT>
<WATER> 43.150 </WATER>
</COMP>
</FOOD>

<FOOD>
<CLASSIF>
<FDA-FFV-8807> J135 M001 P24 N01 A148 B1201 C113 E123 F18 H001 K03
</FDA-FFV-8807>
<USDA-NOB> 1049 </USOA-NDB>
<EUROCODE2> 1.01 </EUROCODE2>
<USDA-NAME> CREAM: FLUID, HALF & HALF, CREAM AND MILK </USDA-
NAME>
</CLASSIF>
<COMP>
<ASH> 0.670 </ASH>
<CHOCDF> 4.30 </CHOCDF>
<ENERC> 545.578 </ENERC>
<FIBC> 0.000 </FIBC>
<PROCNT> 2.960 </PROCNT>
<WATER> 80.570 </WATER>
</COMP>
</FOOD>

<FOOD>
<CLASSIF>
<FDA-FFV-8807> M001 N01 P24 J135 F18 H221 C235 B1201 A182 K03 E123 H208
</FDA-FFV-8807>
<USDA-NOB> 1059 </USDA-NOB>
<EUROCODE2> 1.03 </EUROCODE2>
<USDA-NAME> MILK: FILLED, FLUID, W/BLEND OF HYDR VEGETABLE OILS
</USDA-NAME>

</CLASSIF>
<COMP>
<ASH> 0.800 </ASH>
<CHOCDF> 4.74 </CHOCDF>

<ENERC> 264.280 </ENERC>
<FIBC> 0.000 </FIBC>
<PROCNT> 3.330 </PROCNT>
<WATER> 87.670 </WATER>
</COMP> </FOOD>

<FOOD>
<CLASSIF>
<FDA-FFV-8807> P24 N01 M001 K03 J133 H161 F14 B1201 E106 A148 C235
</FDA-FFV-8807>
<USDA-NOB> 1091 </USDA-NOB>
<EUROCoDE2> 1.3 </EUROCODE2>
<USDA-NAME> MILK: COW, DRY, SKIM, NON-FAT SOLIDS, REGULAR,
WO/ADDED VIT A
</USDA NAME>
</CLASSIF>
<COMP>
<ASH> 7.930 </ASH>
<CHOCDF> 51.98 </CHOCDF>
<ENERC> 1516.368 </ENERC>
<FIBC> 0.000 </FIBC>
<PROCNT> 36.160 </PROCNT>
<WATER> 3.160 </WATER>
</COMP>
</ FOOD>

9. Conversion of data from interchange format

INTRODUCTION

An interchange format file consists of many data items, generally separated by tags but occasionally by whitespace. This stream of data is usually broken up for convenience into lines, but in essence a line break is just more whitespace. In many cases the file will contain data superfluous to one's interests: either foods in addition to those desired or data about unsought food components will be included. The first part of the discussion of extracting data from an interchange file will discuss simple cases of manually finding (with an editor program) certain selected data. The discussion will then progress to more lengthy extractions that might require special programs. The discussion may provide details about the building of such programs, but will discuss what such a program would have to accomplish.

SOME SIMPLE EXAMPLES

Consider a sample task: Find the calcium and iron content of bananas. (This assumes there is a <food> element which has as one of its <BVNAME> elements the content "banana".) The element to be searched for will include a subordinate element <bvname> banana </bvname>. So, by hand or with an editor, one must first find <bvname> banana </bvname> . The banana <food> element probably looks like

```
<food>
...
<classif>
<ifri> ... </ifri>
<bvname> banana </bvname>
</classif>
<fddflt> <meas/> ... </meas/> </fddflt>
<comp> ...
<CA> 5.7 </CA>
. . .
<FE> 63 </FE>
. . .
</comp>
<drvd-comp> ... </drvd-comp>
</food>
```

If the file is positioned at the <food> start-tag preceding <bvname> banana </bvname>, all preceding material can be erased: it will be irrelevant to bananas. Also, all material following the <food> end-tag (</food>) will also be irrelevant and can be erased. Now one can search for <CA> and <FE> without fear of getting a value for the wrong food.

Next, consider this problem: Find all of the names of all of the foods described in an interchange file. First, search for the first <bvname> start-tag and erase it and everything before. Next, find the matching end-tag, </bvname>. Insert a line break and mark the position after the end of the line and before the end-tag, and search for the next <bvname>. Erase everything from the marked position to and including the second start-tag. (Note how this has left the first <bvname> by itself on a line, and the next <bvname> begins the next line.) Now repeat everything from finding the end-tag to erasing up to the next start-tag, over and over until at some point there is no start-tag to be found. At this point, there are no more <bvname>s. Simply erase everything from the last-marked end-tag to the end of the interchange file. What remains is a list of local names, one per line.

How might a program be written to extract local food names automatically? Here is a sample program, written in BASIC.

```
100 if eof#1 goto 200
input #1, dataline$
ndx = index (dataline$, "<bvname>")
if ndx = 0 then goto 100
name$ = sub (dataline$, ndx+12)
ndx = index (name$, "</bvname>")
if ndx> 0 goto 150
120 input #1, dataline$
name$ = name$ + " " + dateline$
if index (dataline$, "</bvname>") = 0 goto 120
ndx = index (name$, "</bvname>")
150 write #2, sub (name$, 1, ndx-1)
goto 100
200 end
```

In this program, input lines are effectively erased by being read but not copied into the output fee. When <bvname> is found, the remainder of the line is copied into "dame$". Subsequent lines are tacked onto "name$" until the end-tag </bvname> is found. Then that part of "name$" prior to the end-tag is written out, and the program returns to skipping lines, looking for the next <bvname>.

The program and the editor algorithm respond differently if a <bvname> is split across two or more lines: the editor algorithm as given above does not include making the name fit entirely on one line while the program does.

COMPLICATIONS

The preceding examples each had a very simplifying aspect. The first was only involved with one <food>; the second, with only one subsidiary element of each <food>. To put it another way, the first looked at several subsidiary elements for each <food> and selected only certain <food>s; the second looked at more than one <food>, but only a single subsidiary element type. In addition, in theory, the <bvname> element could have a different interpretation if it occurred other than immediately subsidiary to <classif>, so that the procedures above would identify some things as food names which were not. Such uses of this particular element are unlikely in practice.

Scanning for Several or Many Components

If more than one subsidiary element is of interest, it is usually necessary to determine the boundaries of each <food> as it is being considered so that the various subsidiary elements are associated with one another and not with the subsidiary elements of another <food> . This suggests that the boundaries of a <food> element must be determined before its content is searched for the subsidiary elements-or at least, when the search is being made linearly topto-bottom, left-to-right, that (l) the <food> start-tag is found first, and (2) as each line is searched for the start-tags of each subsidiary element, the </food> end-tag is also searched for. Note that care must be taken to cover the possibilities that a subsidiary element may occupy more than one line, that another subsidiary element may begin on the same line on which another ends, and that a subsidiary element of the following <food> could possibly occur later on the same line as the current <food>'s end-tag.

Such a parallel search (for several start-tags and an end-tag) requires care in implementing. For example, if the BASIC program of the second problem above were being modified, each line must be checked for every tag of interest; the one to be acted upon must be the one that occurs first. After it is processed, the remainder of the line must be checked for the other tags. (Incidentally, the program as given above made the highly likely but not guaranteed assumption that no two <bvname> elements will fall on the same line.) If it can be guaranteed that every <food> has all of the subsidiary elements, then the end-tag need not be searched for in parallel; indeed if it can be guaranteed (perhaps by a prior sort) that all of the subsidiary elements are not only present but in a prescribed order, then the search for start-tags can be made serially, looking for one only after the preceding one has been found and processed. If a truly parallel search is needed, a text-processing programming language/system which succinctly implements complicated text searches should be considered. Examples of such systems include Digital Equipment Corporation's VAX-TPU^{TM}, the XEDIT system found on IBM's VM/SP system, most versions of the "emacs" editor, and the SNOBOL and ICON languages.

If many, most, or all of the <comp> or <drvd-comp> subsidiary elements are of interest, a variation on this system might be considered. Specifically, when one has found the <comp> start-tag, the next tag should be read no matter what it is. The tag is either a subsidiary start-tag or (after a few repetitions) is the <comp> or <drvd-comp> end-tag. If it is a start-tag, its generic identifier is checked against the list of those elements of interest. If it is one, the data of interest is extracted; otherwise everything up to and including this element's end-tag is erased or ignored. Then the next tag is either a subsidiary start-tag or the <comp> or <drvd-comp> end-tag. The cycle repeats until the end-tag is encountered.

The data selected during this pass can be accumulated in an array, either placed in the proper cell as it is encountered or marked with an identifier indicating which component or derived component it is for and retained for sorting when the <food> is completely read in; in either case, the entire collection of data for that <food> can be written out in the proper order at this time.

Alternatively, the data can be written out as soon as they are encountered, provided that data fields are marked with an identifier that identifies both the food being processed and the component or derived component involved. These output records can then be sorted later at leisure. This is a particularly helpful mechanism for use on very small computers.

Selecting Certain Foods

The problem in scanning a <food> element to determine whether it is to be processed or ignored is that one cannot determine from the start-tag of the element the identity of the food involved. Instead, the food is identified by one or more of the subsidiary data elements. It is absolutely necessary that any potentially useful data that appears in the interchange file before the food identification data be retained in the processing computer until it is determined whether or not this food is of interest. Then that data can be written out or ignored as appropriate.

Given a computer with memory big enough to store all the data of interest for any one <food> as discussed above, it might be easier (though perhaps slower) to read in all of the data of interest and all of the data needed to decide if the food is of interest for each <food> as it is encountered, and then make the decision whether to ignore or erase, or to write out all of that <food>'s data.

OUTPUT FORMATS

Formats for Direct Use by People

If the data values of interest are to be perused only once or twice by a user and then abandoned, the easiest output format available is probably appropriate. Two approaches are prime candidates, with their relative convenience depending upon the strategy for selecting the data of interest from the interchange file.

If the data values are selected using a text processor or editor, it is probably easiest to delete the unnecessary data and leave the remainder of the interchange file in its original format, adding line breaks where necessary to get lines of reasonable length (e.g., at most 65 or 80 characters).

If the data values are selected by copying the desired data to an output file, they may easily be printed in a food-versus-component array: Simply use a format that prints each value in a fixed-width field, with all of the values for a single food printed on one line (presumably preceded by the name of the food). Such output formatting is available in virtually all common computer languages. For example, the FORTRAN format

```
(X,A20,12(X,A4))
```

would handle food names of as many as twenty characters, followed by up to twelve component values of up to four characters each, all printed on an 80-character line.

(The first space character is the FORTRAN carriage control character.)

Other, fancier, formats might be used for inclusion in special reports. These might include column titles, a two-page-wide array, or a non-array-oriented format. In particular, if many items of data about each component are being retained, a "paragraph" of data and descriptive material might be printed for each food. Such a "paragraph" might include various names for the food and statistical information about each primary component datum, along with identifying labels.

Formats for Computer Input

Many computerized food component data bases are maintained as a food-versus-component array of primary component data values, and can be printed out using a format similar to the human-oriented array-based format described just above. A variation on the theme, if the programming language supports variable-length fields for output of data, is illustrated by the following BASIC subroutine:

```
100 FOR I=1 TO 99
WRITE #1, DATA$(I);",";
NEXT I
WRITE #1, DATA$(100)
RETURN
```

In this example, assume that writing out the variable-length string DATA$(I) will be done with the minimum number of characters-no leading blanks or zeroes-and that (as is usual in most BASICs) a semicolon following a WRITE datum suppresses trailing spaces, tabs, or new-lines (record-breaks) that might otherwise be automatically placed after the item. The result is a single record consisting of 100 data items written out in compact form and separated by commas.

Alternatively, the target data base might store the data for a food in several records; for example, in the USDA standard reference data base [35], the data for one food is stored as follows: first, a food name record, with a food-identifying numeral, a type-of-record numeral (000), and a name for the food. Next, several records-one for each reported component with a component-identifying numeral (between 001 and 998), a primary value, and possible secondary values and statistical information. Finally, a record signifying the end of data for this one food, which contains only a type-of-record numeral ("999"). Within each record, the data fields are of constant length; each record is 80 characters, with padding as needed. Such a data collection can easily be written out by a single WRITE instruction to create the name record, a loop of WRITE instructions to emit the associated component records, and a final WRITE instruction for the end-of-food record.

Special Formats for Data Base Management Systems

Many data base management systems (DBMSs) are able to accept a large collection of data at once if the data are provided in some version of one of the array formats described above.

Alternatively, many DBMSs will accept data included in SQL commands [54]. The conversion program would emit the data, extracted from the interchange file, embedded in SQL commands which would direct the creation and initialization of new DBMS records. For example, in an appropriately defined DBMS table, the following SQL commands might add the banana data used in previous examples to a table named FOOD_TABLE:

```
INSERT INTO FOOD_TABLE (LOCAL_NAME, CA, FE)
VALUES ('banana', 5.7, 63)
```

Such a command might be written by a BASIC subroutine such as

```
100 WRITE #2, "INSERT INTO FOOD TABLE (LOCAL_NAME, CA, FE)"
WRITE #2, "VALUES ('"+LOCALNAME$+"', "+CA$+", "+FE$+")"
RETURN
```

where LOCALNAME$, CA$, and FE$ have presumably been given the values "Banana", "5.7", and "63" by another subroutine.

Appendix A registered international food record identifiers

Some International Food Record Identifier components have already been assigned. The first-level facets are listed here for information. More detailed ones are listed as illustrations of the way the IFRI system is expected to work in practice.

Identifier	**Organization or purpose**
UN.	For UN system organizations and, in particular, the FAO-produced regional food tables. Assignments within the "UN" identifier will also be made by the INFOODS Secretariat until another UN agency assumes the responsibility. The stem is followed by an identifier for a UN organization or other body.
UN.FAO.	Food and Agriculture Organization publications and data bases, including the regional food composition tables series. The stem is followed by the identification of a table, publication, or publication group.
UN.FAO.EAsia72	FAO *Food Composition Table for Use in East Asia, 1972*. The stem of the identifier is followed by a numeral, indicating the food number in that table. Example: UN.FAO.EAsia72.250
UN.FAO.NEast82.	FAO *Food Composition Tables for the Near East,* 1982. The stem of the identifier is followed by a roman numeral, a period, and an arabic numeral, indicating the table number and the food number in that table. Example: UN.FAONEast82.III.23.
UN.UNU.	United Nations University publications, data bases, or food tables.
UN.WHO.	World Health Organization publications, data bases, etc.
OC.	Oceaniafoods

Appendix B element registration form

INTRODUCTION

This appendix contains a sample form for registering a new element, as discussed in Chapter 7. Updated versions of the form will be supplied to regional data centres as needed.

A complete registration application must include:

- Part I
- either Part II or Part III (see instructions, below)
- Part IV

As discussed in Chapter 7, existing element contents may sometimes be extended. A proposal to extend an existing element must demonstrate a thorough understanding of the format and content of that element. Such a proposal must justify the extension thoroughly, including providing documentation in support of the extension and explaining why an existing element should be extended rather than registering a new element.

PART I: APPLICANT INFORMATION

Date of Submission:	
Name:	Title:
Organization:	

Address (include country and INFOODS region):

Telephone:	Telex:
Cable:	Fax:

Electronic mail address (with network and routing information as defined for the <email/> element):

Proposed Element or Modification Category

Only elements subsidiary to <food> or <header> elements, or their subsidiary elements, may be registered. Indicate whether this application proposes (select one):

- to register an element of the subsidiary elements to <food>, i.e., subsidiary to <classif>, <comp> or <drvd-comp> or their contents (complete Part III).
- to register an element of the subsidiary elements to <header>, i.e., subsidiary to <sender> or <source> or their contents (complete Part II). (See Chapter 7 for an explanation of the registration process.)

Definition and Justification

Define and justify the proposed element in the context of the interchange system and its existing elements:

Secretariat/Registration Authority Use Only
Received:
Action:

PART II: HEADER ELEMENT PROPOSAL

The proposed element is immediately subsidiary to (select one):

○ the <sender> element
○ the <source> element

The proposed tag for the element is: <________>.

PART III: FOOD ELEMENT OR COMPONENT PROPOSAL

The proposed element is immediately subsidiary to (select one):

○ the <classif> element
○ the <comp> element
○ the <drvd-comp> element

The proposed tag for the element is: <_________>.

Units of Measure (required)

For elements subsidiary to <comp>, describe the common or default unit of measurement for expressing the quantity per 100g of edible portion of food:

○ milligrams
○ other-Unit:__________

If a unit of measure other than milligrams is used, explain:

For elements subsidary to <drvd-comp>, describe the common or default unit of measurement for expressing the quantity per quantity of food component and identify the component (for example, "mg/g nitrogen" or "mg/100g protein"):

Subsidiary <Units/> Element (required)

The proposed element (select one):

○ permits the use of the <units/> element
○ prohibits the use of the <units/> element

Synonyms (required)

Identify synonymous names by which the food component or derived component is known (including common abbreviations):

Data Tables and Literature (required)

If data on the specific food component or derived component now appear in data bases, tables, or other literature sources, or are expected to appear in tables, articles, or data bases in preparation, provide a reference for (at least some of) those tables. If the component does not appear in food composition data bases, tables, or literature sources, explain why the proposed element should be registered:

Analytic Methods

Identify literature references' for any specific methods of analysis or computation associated with the proposed element:

Existing Elements

If the proposed element concerns a component or derived component which has already been registered in the interchange system (according to a different analytic method or some other criterion), identify the features of the proposed element distinguishing it from the existing element(s) for the component or derived component in question:

PART IV: CONTENT DESCRIPTION FOR ALL ELEMENTS

The proposed element's content is composed of (select the one which is applicable):

- one or more data values only (complete subsection A).
- one or more keywords only (complete subsection B).
- data values and keywords only (complete both subsections A and B). For data-and-keywords contents, text data is prohibited.
- subsidiary elements only (complete subsection C).
- a combination of data values and/or keywords and subsidiary elements (complete subsections A, B. and C).

A. Data

How many data values are: Data values are:

○ required: ○ optional:	○ ordered ○ unordered

Data values may be (select one):

○ text values
○ mixed text and/or numeric values
○ numeric values

Explain the meaning of the data values. If any values are optional, explain what their absence means. Provide any additional information which would contribute to proper interpretation of the values:

B. Keywords

Provide a list of valid keywords and their meanings:

Should additions to this list be permitted in the future?

○ Yes
○ No

If YES, define the rules for keyword extension. If NO, explain briefly why not:

C. Subsidiary Elements

Define all subsidiary elements. (If the subsidiary element already exists in the interchange system [e.g., any element listed in this manual, such as <cmt/>], or is specific to the proposed element only, it must be identified here. If the subsidiary element does not already exist and could be used more generally in the interchange system, it should be listed here and a separate registration application included with this application.):

Cross-references

List and explain any cross-references to other elements:

Additional Information

List any relevant additional information, if any:

Example

Provide an example of a complete proposed element. If the element may contain optional data fields, please give several examples and explain their interpretation:

Glossary

ANSI

The American National Standards Institute, the body responsible for ratification and distribution of standards in the United States.

alphanumeric

Consisting of alphabetic and numeric symbols.

closing delimiter

A character or tag that indicates where a data field or element ends. Usually matched to a similar "opening delimiter", e.g., the symbol " is typically used as a closing delimiter for a data field (which is typically called a quoted string or a quotation) that begins with the opening delimiter ".

content

The content of an element may consist of one of three forms: (1) data alone with no subsidiary elements (data content), (2) data followed by one or more elements (mixed content), or (3) one or more elements without preceding data (element content).

Element with data content:

```
<VITB12> 1.2 </VITB12>
```

Element with mixed content:

```
<NA> 0.12 <unit/> MMOL </unit/> </NA>
```

Element with (multiple) element content:

```
<food>
<classif>
<ifri> food_record_identifier </ifri> </classif>
<meas> measurement elements </meas>
<comp> food component data elements </comp>
<drvd-comp> derived food component and descriptive data elements
</drvd-comp> </food>
```

conversion specification file

A file that specifies the conversion process between the data formats of a particular system and that of another. It typically includes field locations and lengths and an indication of field content (e.g., particular food components represented).

cross-references

In cases where a tag is defined in terms of one or more other tags (for example, "similar to another tag but slightly different," "defines a subset of what another tag defines," "refers to the same substance but by a different analytical method," and so on), those other tags are termed cross-references.

data value

A single numeral or string representing the value for a particular field (e.g., a nutrient) for a particular entity (e.g., a food). Sometimes called a "datum".

edible portion

The fraction of a food or food product typically eaten and on which analyses are usually based. The perception of what is edible can differ from one culture to another, so the edible portion should be carefully described.

element

An element consists of a start-tag followed by its content followed, if the start-tag requires it, by a matching (i.e., same generic identifier) end-tag. See examples under "content", above, and the discussion in Chapter 3.

end-tag

An end-tag is a tag that marks the end of an element and is preceded by the content of that element. By convention, an end-tag has the form <generic identifier>. See "start-tag" and "generic identifier" in this section, and Chapter 3.

If the start-tag of an element is <food> the end-tag is </food> . Likewise, if the start-tag of an element is <unit/> the end-tag is </unit/>.

extensible

An element is described as extensible if, upon sufficient justification, additions can be made to the keywords or elements that can be used in its content.

fixed-field system

A system for organizing data such that each item occupies a preset, and universally agreed upon, number of columns. Each item is located by measuring off a fixed distance-determined by the number of characters in each of the preceding fields- from the beginning of the record. These systems can be made quite efficient from a programming standpoint and are easy to program. They do require that blank space be left for all nutrients that are not supplied, so the number of characters wasted will be very large when various data are not available.

formatted data

Data content associated with a particular element that must appear in some particular form and order. In general, the data content is everything appearing between a start-tag and the corresponding end-tag or, for elements that have a start-tag but no end tag, the next tag in sequence. If the data are formatted, the description of the element will specify exactly what may appear, and in what order. The alternative is "free text" (see below), also called "unformatted data" or "unformatted text".

free text

Text, consisting of alphabetic, numeric, and punctuation characters, that is not restricted as to format or structure. The usual alternatives are numeric values, keywords, and elements. In the context of the interchange system, free text is usually referred to as "unformatted data" or "unformatted text".

gateway

Computer software, or the machine on which it runs, that links facilities such as networks with different protocols and performs translations among them. A "mail gateway" is one that converts electronic mail and address formats between one network and another.

generic identifier

That pan of a tag which is enclosed between the opening " <" or " </" and the closing "> ", exclusive of qualifiers such as the "85" in <infoods 85>, is termed the generic identifier. The generic identifiers of the start-tag and end-tag of an element are identical. For the tag <food>, the generic identifier is "food". For the tag <unit/>, the generic identifier is "unit/". For the tag <infoods 85>, the generic identifier is "infoods".

ifri

The international food record identifier, a regionally assigned identification code for food data records (tables or data bases).

immediately subsidiary

An element or data value is described as "immediately subsidiary" to another one when there are no intervening nested elements. In "<xx> A <yy> B </yy> </xx>", A and <yy> are immediately subsidiary to the <xx> element, and B is immediately subsidiary to the <yy> element, but, while B is subsidiary to the <xx> element, it is not immediately subsidiary, since <yy> intervenes.

A given data value or element can be immediately subsidiary to only one element.

interchange format

The actual structure of a data file used in INFOODS data interchange. One component of the overall "interchange system", where other components include the "tags" or generic identifiers that identify various pieces of information, the conventions for locating and requesting data files, the mechanisms for assigning international food record identifiers, and the computer programs and operational arrangements for regional data centres.

ISO

The International Organization for Standardization, the body responsible for evaluating and setting standards internationally.

left-justify

See right-justify.

keyword

A word, acronym, or other short sequence of characters that is chosen from a restricted list and that has specially defined meaning when used in context. See Chapter 7.

macro

A sequence of commands to be applied by some process, typically invoked by a single command or the definition for a (possibly complex) abbreviation. The term is used in the context of text processing to describe the text to be substituted for some other text (usually repeatedly) or the instructions for making the substitutions.

metadata

Where "data" contain information about some topic, the term "metadata" is used to denote data about the data, including how they were obtained, their statistical properties, and special circumstances affecting them.

nesting

When elements are embedded in the scope or range of other elements, they are said to be nested within the ones in which they are embedded. The depth of the embedding is sometimes referred to as the "nesting level" of the elements. For example, if we have:

<PROCNT> 3.3 FAO 638 <cmt/> Note USDA values for same. </cmt/> </PROCNT>

we would describe the <cmt/> element as being nested within the <PROCNT> element. The depth of nesting has an effect on the sophistication of the computer software required to process the elements. See "immediately subsidiary", above.

opening delimiter

See "closing delimiter", above.

repeatable

The term "repeatable" is applied to an element immediately subsidiary to another element. It means that the given element may occur more than once as an immediate subsidiary to the other element.

right-justify

In a field of prespecified length containing text, when significant information is placed to the right end of the field it is right-justified. Le justification involves putting the information to the left end of the field. These two notions provide for a distinction that is very significant to computers, even if not to people.

" right-"
" justification"

"left- "
"justification "

SGML

The Standard Generalized Markup Language, the language for structuring text upon which the Interchange Format was designed. It is specified in International Standard ISO 8879 [53]

solidus (slant, or slash)

The solidus, /, identifies an end-tag (</generic identifier>), or a start-tag which requires an end-tag (<generic identifier/>). The term "solidus" is used interchangeably with "slant" and "slash".

start-tag

A start-tag is a tag that marks the beginning of an element and is followed by the content of that element. By convention, a start-tag has the form <generic identifies>. See "end-tag" and "generic identifier", above. <Food> and <unit/> are examples of start-tag

structural element

Structural elements determine the ordering of elements in an interchange file, and define the form, or structure, of that file. Their content consists of one or more subsidiary elements only and no data. The <header>, <sender>, <source>, <food>, and <classif> elements, for example, are all structural elements.

subsidiary element

Any element which forms pan or all of the content of another element (see "immediately subsidiary" and "nesting").

tag

In SGML, the particular symbols used to mark up a document and identify its components are called tags. In the system outlined in this memo, the tags correspond to the names for fields in the interchange file.

tagname

The term "tagname" refers exclusively to the text portion of a generic identifier, exclusive of the possible terminating slant. Although the term "tagname" has been used extensively in certain INFOODS documents, it is not interchangeable with "generic identifier".

unformatted data

Data, usually a string of characters, which may contain whitespace characters. The beginning and end of an unformatted data string are typically marked by tags or by some other type of opening and closing delimiters. Equivalent to "free text" and "unformatted text".

value

An informal term for "data value", qv.

whitespace

A character, or sequence of characters, that are used to separate keywords, numerals, or elements. This term is used because these characters appear as spaces, or sequences of spaces, on the printed page. In computer character coding terms, the whitespace characters are space, horizontal tab, vertical tab, and the new line sequence.

Bibliography

BOOKS AND ARTICLES

1. Arab L, Wittier M, and Schettler G. *European Food Composition Tables in Translation*. Berlin and Heidelberg: Springer-Verlag, 1987.

2. CompuServe. Graphics interchange format, version 89a. Columbus, Oh., USA: CompuServe, Inc., 1990.

3. Coombs JH, Renear AH, and DeRose SJ. Markup systems and the future of scholarly text processing. *Communications of the ACM*, 30, 11 (November 1987).

4. Cubitt RE. Meta data: An experience of its uses and management. *Proceedings of the Second International Workshop on Statistical Database Management*, Lawrence Berkeley Lab, Berkeley, Calif., USA, 1983, pp. 280-286. Available through NTIS.

5. Dawson R. Klensin JC, and Yntema DB. The Consistent System. *The American Statistician*, 35, 3 (August 1980), pp. 169-176.

6. Efron B and Gong G. A leisurely look at the bootstrap, the jackknife, and cross-validation. *The American Statistician,* 37, 1 (February 1983), pp. 36-48.

7. Efron B and Morris C. Data analysis using Stein's Estimator and its generalization. *J Amer Statistical Assn*, 70, 350 (June 1975), pp. 311-319.

8. Fisher, RA. *Statistical Methods for Research Workers*. 14th edition. London: Collier-Macmillian, 1970.

9. Food and Agriculture Organization. *Food Composition Tables for Use in East Asia*. Rome: UN Food and Agriculture Organization, 1972.

10. Food and Agriculture Organization. *Food Composition Tables for the Near East*. Rome: UN Food and Agriculture Organization, 1982.

11. Greenfield H and Southgate DAT. *Guidelines for the Production, Management, and Use of Food Composition Data*. In preparation.

12. Heintze D, Klensin JC, and Rand WM. *International Directory of Food Composition Tables*. MIT, Cambridge, Mass., USA: INFOODS Secretariat, 2nd edition, September 1988.

13. Food and Drug Administration. Factored Food Vocabulary. As presented at the NCI Food Data System-Factored Food Vocabulary (FFV) Workshop, June 4-5, 1987.

14. Hoaglin DC, Mosteller F. and Tukey JW. *Understanding Robust and Exploratory Data Analysis*. New York: John Wiley & Sons, Inc., 1983.

15. Howson C and Urbach P. *Scientific Reasoning: The Bayesian Approach*. La Salle, Illinois: Open Court, 1989.

16. Klensin JC. A statistical database component of a data analysis and modelling system: Lessons from eight years of user experience. *Proceedings of the Second International Workshop on Statistical Database Management*, Lawrence Berkeley Lab, Berkeley, Calif., USA, 1983, pp. 280-286. Available through NTIS.

17. Klensin JC, Feskanich D, Lin V, Truswell AS, and Southgate DAT. *Identification of Food Components for INFOODS Data Interchange*. Tokyo: United Nations University, 1989.

18. Klensin JC and Romberg RM. Statistical data management requirements and the SQL standards: An evolving comparison. In Rafanelli M, Klensin JC, and Svensson *P. Statistical and Scientific Database Management: Fourth International Working Conference.* Lecture Notes in Computer Science No. 339. Berlin and Heidelberg: Springer-Verlag, 1989.

19. Klensin JC. Regional center design and requirements. Paper presented at the INFOODS Policy Committee meeting, Budapest, November 1987. Abridged version available from INFOODS Secretariat as INFOODS IS/N7 (See reference 56).

20. Laboratory of Architecture and Planning. *Consistent System: Janus Reference Manual*. Cambridge, Mass., USA: MIT Laboratory of Architecture and Planning, 1980 (periodically revised).

21. Moran R. ed. *SQL*Plus Report User's Guide*, Version 1.0. Belmont, Calif., USA: Oracle Corporation, 1985.

22. Mosteller F and Tukey JW. *Data Analysis and Regression*. Reading, Mass., USA: Addison-Wesley, 1977.

23. Paul A and Southgate DAT. *McCance & Widdowson's The Composition of Foods*. London: HMSO, 1978.

24. Rand WM, Pennington JAT, Murphy SP, and Klensin JC. *Compiling Data for Food Composition Data Bases*. Tokyo: United Nations University Press, 1992.

25. Rand WM and Pennington JAT. Nutrient variability and reliability: What to put in a food table?. *Proceedings of the 16th National Nutrient Databank Conference, 1991*. Ithaca, NY, USA: The CBORD Group, 1992.

26. Sachs J. *SQL*Plus Reference Guide*, Version 2.0. Belmont, Calif, USA: Oracle Corporation, 1986.

27. Shoshani A and Wong HKT. Statistical and scientific database issues. *IEEE Transactions on Software Engineering*, SE-11(10):1040-1047, October 1985.

28. Snedecor GW and Cochran WG. *Statistical Methods.* 7th edition. Ames, Iowa, USA: Iowa State University Press, 1980.

29. Stamen JP and Wallace R. Janus: A data management and analysis system for the behavioral sciences. *Proceedings of the 1973 Annual Conference of the ACM*, New York: Association for Computing Machinery, 1973.

30. Stewart KK. Editorial: Are they different. *J Food Comp and Anal*, 1, 2 (March 1988), p. 103.

31. Truswell AS, Bateson D, Madafiglio D, Pennington JAT, Rand WM, and Klensin JC. Facets for the description of foods: INFOODS guide to the aspects of foods which should be considered when describing them for a food composition database. INFOODS Working Paper, July 1987.

32. Truswell AS, Bateson D, Madafiglio D, Pennington JAT, Rand WM, and Klensin JC. INFOODS guidelines for describing foods: A systematic approach to describing foods to facilitate international exchange of food composition data. *J Food Comp and Anal*., 4, 1 (March 1991), pp. 18-38.

33. Truswell AS, Bateson D, and Madafiglio K. Manual to accompany a scheme for naming and describing foods in food composition tables and data bases. INFOODS Working Paper, November 1986.

34. Tukey, JW. *Exploratory Data Analysis*. Reading, Mass., USA: Addison-Wesley, 1977.

35. US Department of Agriculture. *Composition of Foods*: Raw, Processed, Prepared Agriculture Handbook No. 8, revised. Washington, DC: Science and Education Administration, US Department of Agriculture, 1976-1990.

AMERICAN NATIONAL STANDARDS INSTITUTE DOCUMENTS

These documents are, in most cases, equivalent to and redundant with the International Standards listed below. They can be ordered from the Order Department, American National Standards Institute, 11 West 42nd St., New York, NY 10018, USA. Telephone + 1 212 642 4900.

36. ANSI X3.42-1975: American National Standard for the Representation of Numeric Values in Character Strings for Information Interchange.

37. ANSI X3.30-1985: American National Standard for the Representation of Calendar Date and Ordinal Date for Information Interchange

38. ANSI X3.135-1989: American National Standard for Information Processing Systems-Database Language-SQL with Integrity Enhancement.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION DOCUMENTS

These documents are, in general, available from the sales offices of national standards organizations (the "member bodies") in each country. If a country does not have a member body, or the member body cannot be located, inquiries should be addressed to International Organization for Standardization, Central Secretariat, 1, rue de Varembé, Case postale 56, CH-1211 Genève 20, Switzerland.

39. ISO 639: International Standard-Code for the representation of names of languages.
40. ISO 646: International Standard-7-bit coded character set for information interchange. 1983

41. ISO 2014: International Recommendation-Representation of calendar date and ordinal date for information interchange.
42. ISO 2108: Documentation-International Standard Book Numbering (ISBN). 1978.
43. ISO 3166: International Standard-Codes for the representation of names of countries.
44. ISO 3297: Documentation-International Standard Serial Numbering (ISSN). 1986.
45. ISO 6093: International Standard-Information processing-Representation of numerical values in character strings for information interchange. 1985.
46. ISO 6523: Structure for the Identification of Organisations. 1984.
47. ISO 8859: International Standard 8 bit single byte coded graphic characters
48. ISO 8859-1: International Standard~bit single-byte coded character sets-Part 3: Latin alphabet no. 1. 1987.
49. ISO 8859-5: Draft International Standard-bit single-byte coded character sets-Part 5: Latin/Cyrillic alphabet.
50. ISO 8859-6: International Standard-bit single-byte coded character sets-Part 6: Latin/Arabic alphabet. 1987.
51. ISO 8859-7: International Standard-bit single-byte coded character sets-Part 7: Latin/Greek alphabet. 1987.
52. ISO 8859-8: International Standard-bit single-byte coded character sets-Part 8: Latin/Hebrew alphabet. 1988.
53. ISO 8879: International Standard-Information processing-Text and office systems-Standard Generalized Markup Language (SGML).
54. ISO 9075: International Standard Database Language SQL. 1987.

INFOODS INFORMATION SYSTEMS WORKING PAPERS

These working papers document the development history of the interchange system and its various components. This manual is intended to replace them, and contains information which is more complete and fully developed. The original working papers do, however, contain more discussion about the reasons for particular decisions than the present manual.

55. INFOODS/IS N6: Interchange standard: draft for review
56. INFOODS/IS N7: Regional food composition data centers/systems
57. INFOODS/IS N15: Initial root (structural) tag list
58. INFOODS/IS N16: Requirements for registering a tag
59. INFOODS/IS N17: Additional discussion of the interchange scheme
60. INFOODS/IS N20: Food description, automatic coding, and retrieval
61. INFOODS/IS N21: Introduction to the interchange scheme conversion programs
62. INFOODS/IS N22: Representation of trace, missing, and zero values
63. INFOODS/IS N30: Tags for local names and classifications
64. INFOODS/IS N31: Comments and response on IS N30
65. INFOODS/IS N32: A new structural tag category-derived measures
66. INFOODS/IS N35: Statistical tags
67. INFOODS/IS N36: Producing interchange files from database systems