ORNL/TM-12141

ornl

OAK RIDGE NATIONAL LABORATORY

MARTIN MARIETTA

Feasibility Study—Computerized Application of the Hazardous Material Regulations

J. J. Ferrada
R. R. Rawl

MANAGED BY
MARTIN MARIETTA ENERGY SYSTEMS, INC.
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

ORNL/TM-12141

Chemical Technology Division

Feasibility Study—Computerized Application of the Hazardous Material Regulations

J. J. Ferrada
V. M. Green
R. R. Rawl

Published: September 1992

Prepared for the
Office of Environmental Restoration and Waste Management
(EM 50 10 01 0)

Prepared by the
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831
managed by
MARTIN MARIETTA ENERGY SYSTEMS, INC.
for the
U.S. DEPARTMENT OF ENERGY
under contract DE-AC05-84OR21400

Contents

LIST OF ACRONYMS

AI	Artificial Intelligence
AMTS	Automated Transportation Management System
BGI	Borland Graphics Interfaces
CRADA	Cooperative Research and Development Agreement
DOE	Department of Energy
DOT	Department of Transportation
ECA	Engineering Coordination and Analysis
IHS	Information Handling Services
LSA	Low Specific Activity
OOP	Object-Oriented Programming
ORNL	Oak Ridge National Laboratory
OSHA	Occupational Safety and Health Administration
PC	Personal Computer
PDC	Prolog Development Center
RAM	Radioactive Material
RCRA	Resource Conservation and Recovery Act
SARA	Superfund Developments Reauthorization Act
SCO	Surface Contaminated Object
TSCA	Toxic Substance Control Act

FIGURES

FEASIBILITY STUDY—COMPUTERIZED APPLICATION OF THE HAZARDOUS MATERIAL REGULATIONS

J. J. Ferrada

R. R. Rawl

ABSTRACT

The feasibility of developing a full expert system for transportation and packaging of hazardous and radioactive materials was initiated within the framework of three subtasks: 1) analysis of commercial packages related to regulation scanning, 2) analysis of computer languages to develop the expert system, and 3) development of expert system prototypes. The strategy to develop the latter subtask was to first, develop modules to capture the knowledge of different areas of transportation and packaging and second, to analyze the feasibility of appending these different modules in one final full package. The individual modules development contemplated one prototype for transporting and packaging of radioactive material and another for transporting hazardous chemical materials. In the event that it is not feasible to link these two packages, the modules can always be used as stand-alone tools, or linked as a single package with some restrictions in their applicability. The work done during this fiscal year has focused on developing a prototype for transporting radioactive materials.

The analysis of the commercially available software, RegScan and Environmental/Safety Library indicated that both packages, although very useful for navigating the pertinent regulations, are not particularly suitable for the determination of the type of packages required for hazardous and radioactive material transportation. However, the regulations can be downloaded from these packages and used as the source data base for the regulations that will be accessed by the expert system.

From the analysis of the implementation of the different features required for the expert system prototype, it was concluded that the developmental efforts should be directed to two versions of the prototype. First, a simpler DOS version that could be run on any computer machine to accelerate the review process was recommended. Second, a more sophisticated Windows Hypermedia version that included every required feature was built as a proof-of-concept. The DOS version of the prototype was developed using PDC Prolog because it allows more flexibility with the different incorporated features. The hypermedia version was developed using a combination of GUIDE and Prolog with very good results.

GUIDE does not have an incorporated rule-based system that allows the development of expert systems. However, LOGiiX (a GUIDE embedded language) provides commands that can access rule reasoning mechanisms at any point during the consultation.

The first phase in demonstrating the feasibility of developing an expert system prototype has produced interesting results. It is possible to represent knowledge about radioactive material transportation packaging using a logic diagram that has been translated into a rule-based system. The features required to assist a user during a consultation session have been implemented in both DOS and Windows 3.0 environments. The mechanism to download updated regulatory information from RegScan has proven to be technically feasible and within the purview of the license agreement. The activities related to the assessment of the required review and the validation and verification processes for the operational use of the radioactive material transportation prototype is in the planning stage. Links with the Automated Transportation Management System (ATMS) will begin. One of the subtasks proposed for FY 1993 is the investigation of possible ties of the transportation packaging expert system with ATMS. The knowledge gained from this study will indicate the type of tools and mechanisms necessary to link the expert system with the ATMS.

The second phase of the demonstration will include the feasibility analysis of the hazardous chemical materials transportation packaging expert system prototype. The possibility of developing an expert system for the transportation packaging of hazardous chemical materials to be linked to the radioactive material transportation packaging expert system will be evaluated. The analysis will determine the possibilities of linking the program to the ATMS.

1. INTRODUCTION

Artificial intelligence (AI) is a growing field that is making slow but steady progress in various engineering applications. Engineers are directly involved in using AI to develop expert systems that will model a wide variety of process systems. Correctly applied, AI techniques enable the development of an expert system that is capable of representing and manipulating expert knowledge at the same level as a human expert. An expert system emulates human expertise by applying the techniques of logical inference to a knowledge base. Thus, an expert system contains three components, (1) a knowledge base that contains all of the available information about a specific problem, (2) an inference mechanism that normally performs reasoning by referring to a set of rules for using that knowledge to resolve a given situation, and (3) a user interface that accesses the necessary information.

The ability of the computer to simulate and make decisions as a subject-matter expert has greatly increased potential applications. With this type of computing power available in increasingly easy-to-access format (through object-oriented programming and hypermedia), the development of an expert system based on hazardous materials regulations is possible.

One limitation that has precluded the development of such a system for hazardous material transportation is the data base that is needed to support it (i.e., the regulations have not been available in a machine-usable format). Now that the hazardous materials regulations are commercially available in PC-compatible formats, the door to developing such a system has been opened. It may be possible to develop a system that can take initial input (e.g., characteristics of the material being shipped) and provide the user with the information necessary to expedite the following decisions:

- making a packaging selection;
- guiding the loading of the packaging for shipment;
- preparing the shipping papers or hazardous waste manifest by providing information such as shipping name, hazard class, packaging group, and whether the material constitutes a reportable quantity or an inhalation poison;
- identifying further restrictions, such as prohibitions of shipping on a passenger aircraft; and
- providing additional hazard communication information, such as proper marking and labeling for the package and placarding for the transport vehicle.

Ultimately, it could be possible to generate all of the shipping paperwork this way. The expert system would ensure that even the most obscure details of making unusual shipments are readily available to all users, thus emulating an expert quite effectively and thereby significantly reducing the likelihood for human error. Broad capability systems [e.g., Automated Transportation Management System (ATMS)] as well as personal computers could be used to provide access to the system.

Under the sponsorship of the Department of Energy's (DOE's) Office of Environmental Restoration and Waste Management (EM-561), the Transportation Technologies Group of the Engineering Coordination and Analysis Section (EC&A), which is part of the Chemical Technology Division at Oak Ridge National Laboratory (ORNL) has investigated the design and development of a computerized prototype application of hazardous materials transportation regulations. The objective of this task was to evaluate the feasibility of developing a computerized expert system that will ensure straightforward, consistent, and error-free application of the hazardous materials regulations. The task entailed the analysis of what an expert in hazardous materials shipping information could/should do. In addition, the analysis included determining the status of commercially available computer software that performs similar functions. This information, in conjunction with existing computer tools, was used to determine the requirements to produce the expert system and served as the basis to develop preliminary prototype modules of the system.

1.1 STRATEGY

A structured approach has been used to determine the feasibility of developing the expert system. These steps included:

1. Identification of the system requirements (what it is designed to do, constraints, etc.). These requirements consider interactions with potential users.

2. Identification of existing resources and efforts that may support an expert system. Examples of identifiable support include evidence of similar development in industry, industrial interest in Cooperative Research and Development Agreements (CRADAs), identification of similar efforts underway within DOE, investigation of licensing implications for using commercial data bases, etc.

3. Interaction with transportation experts to determine the most appropriate logic diagram for packaging activities. The resulting logic diagram for radioactive material shipments was the source of knowledge acquisition for the expert system development.

4. Development of the proof-of-concept for the system using state-of-the-art software, such as object-oriented programming (OOP) and hypermedia, which will make future development much easier and less expensive. The proof-of-concept was obtained by developing a prototype module of the expert system for radioactive materials transportation packaging. The prototype development will eventually indicate the procedure of how the full expert system should be developed for all hazardous material shipments.

5. Verification/validation of the expert system prototype. The prototype must be submitted for review by experts in the radioactive materials transportation field. Their suggestions and comments will influence the final features of the expert system.

1.2 EVALUATION OF COMPUTERIZED TOOLS

The evaluation of computerized tools was a two-part task. First, a measure of the existing commercial computer programs that assist in transportation activities was conducted. Secondly, the available computing tools for development of the expert system were analyzed. Detailed results of the analysis of these two subtasks are included in Sect. 2.

1.3 EXPERT KNOWLEDGE ACQUISITION

The development of an expert system in any developmental stage requires a close relationship between the developer and the subject expert. Accordingly, the knowledge must flow from the expert to the developer[1] in such a manner that the latter can visualize the "branches" and "trunk" of the required knowledge "tree." The expert system developer must 0analyze the problem and select a portion on which to focus. Typically, the initial prototype is concerned with only a particular portion of the problem and will not provide the full range of ultimate solutions.

Attempts have been made by Tarnuzzer[2] to incorporate the transportation packaging knowledge base for radioactive materials into a logic diagram. This logic diagram represents the decision-making process that the user may follow when evaluating the transport of radioactive materials. In essence, the logic diagram must represent a structure of pathways throughout the regulations relevant to radioactive materials transportation packaging. Anticipated HM-169A regulatory changes have been included in the prototype, which required some changes to the original logic diagram. Details of the knowledge acquisition process are described in Sect. 3.

1.4 PROTOTYPE DEVELOPMENT

The knowledge extracted from the expert has to be transformed to a rule-based system. This transformation constitutes only a portion of the activities involved in the prototype development because there are other elements of the program that require analysis and testing. User interaction and required features are two relevant characteristics of any expert system. Once a decision has been made about these aspects, the selection of computer tools needs to be addressed. During the development of the prototype, several programming computer tools have been analyzed. Among these tools, AI languages (such as Prolog and Smalltalk), object-oriented languages (such as C++), and multimedia programs (such as GUIDE), have all proven very promising. Details of the analysis and selection of computer languages for the prototype development are included in Sect. 4.

2. EVALUATION OF COMPUTERIZED TOOLS

The evaluation of computerized tools consisted of two parts. First, a measure of the existing commercial computer programs that assist in transportation activities was conducted. Secondly, the computer tools for development of the expert system application were analyzed.

2.1 STATUS OF RELATED COMMERCIALLY AVAILABLE COMPUTER PROGRAMS FOR HAZARDOUS MATERIALS TRANSPORTATION

There are a few computer systems on the market that assist the user in decision making for transportation and packaging activities. It is necessary to determine whether these systems effectively answer all of the requisite questions for hazardous/radioactive materials transportation problems. Consequently, before any attempt was made to develop an expert system prototype, the available commercial packages needed to be thoroughly scrutinized.

The evaluation of commercially available programs began with systems involving hazardous materials regulations on personal computer (PC)-compatible formats. Two demonstration diskettes related to regulations and requirements for shipping hazardous wastes were obtained—RegScan produced by Regulation Scanning™ and the Environmental/Safety Library from Information Handling Service (IHS). The results of the demonstrations are described in the following sections.

2.1.1 RegScan*

RegScan is a menu-driven application that allows the user to browse through the regulations with a search word/phrase mechanism that locates portions of the regulations that deal with the subject to be investigated. One of the interesting features of RegScan is the forward referencing mechanism that allows the user to follow a path of references

*RegScan is produced by Regulation Scanning™, Williamsport, Pennsylvania.

through the regulations. Another feature of RegScan is the display of the complete entry in the hazardous materials table for the material that the user has selected. Among the limitations found in RegScan are the inability to illustrate complex tables and figures and the lack of additional information, (e.g., explicit additional information on restrictions or prohibitions in shipping, or explicit mechanisms to find these issues).

As a menu-driven application, RegScan is able to take the users through different hazardous materials transportation topics. The **Main Menu** is the starting point when using RegScan. This menu provides access to RegScan's features from the list of options on the right side of the **Table of Contents** menu items (Fig. 1). The function of the **Main Menu** is to help the user locate needed information within the various titles and parts of the Code of Federal Regulations that comprise the data base of the RegScan module. Option 2 of the **Table of Contents** is **UN/NA Number**. If the user, for example, enters in UN2080, the substance which has that identification is shown as illustrated in Fig. 2.a (acetyl acetone peroxide in this case). Then the description of the substance is illustrated in Figs. 2.b and 2.c.

Another search procedure of RegScan uses the proper shipping name. Figure 3.a illustrates the process of searching for acetone. Figure 3.b illustrates a resulting table with hazard class and identification number. Figure 3.c illustrates the hazardous materials description and proper shipping name for acetone. Figure 3.d shows the current regulation 49 CFR 173.118 (Code of Federal Regulations) for limited quantities of flammable liquids.

Option 3 of the **Main Menu** allows scanning of different tables as shown in Fig. 4.a. Figure 4.b illustrates one search run for isotopes in which a list of isotopes is given to the user (Fig. 4.c).

```
                          RegScan 49 Main Menu
Reg. No. 017C00-4C02A-C056B                         Updated thru Sep 01 1990

       1. Proper Shipping Name          a. Table of Contents
       2. UN/NA Number                  b. Part/Section/Subsection
       3. Table Scan Menu               c. Word/Phrase Search
       4.                               d. Print Section to File
       5.                               e.
       6. Exit to DOS                   f. Aux. Services Menu

       Copyright (c) Regulation Scanning Technology Corp. 1987,1988
```

Please enter the Proper Shipping Name as it appears in the Table in 172.101

Fig. 1. RegScan main menu.

```
                          RegScan 49 Main Menu
Reg. No. 017C00-4C02A-C056B                    Updated thru Sep 01 1990

       1. Proper Shipping Name          a. Table of Contents
       2. UN/NA Number                  b. Part/Section/Subsection
       3. Table Scan Menu               c. Word/Phrase Search
       4.                               d. Print Section to File
       5.                               e.
       6. Exit to DOS                   f. Aux. Services Menu

       Copyright (c) Regulation Scanning Technology Corp. 1987,1988
```

```
Please enter the UN or NA number preceded by UN or NA as appropriate
UN2080
```

Fig. 2.a. Search by UN/NA number.

(2) Proper Shipping Name	(3) Hazard Class	(3A) ID Number
Acetyl acetone peroxide , [in solution with not more than 9% by weight active oxygen. See] Organic peroxide, liquid [or] solution, n.o.s.		UN2080
Acetyl benzoyl peroxide , [not more than 40% in solution]. [See] Acetyl benzoyl peroxide solution, [not over 40% peroxide]		UN2081
Acetyl benzoyl peroxide solution , [not over 40% peroxide]	Organic peroxide	UN2081
Acetyl cyclohexanesulphonyl peroxide , [not more than 82%, wetted with not less than 12% water. See] Organic peroxide, solid, n.o.s.		UN2082
Acetyl cyclohexanesulphonyl peroxide , [not more than 32% in solution. See] Organic peroxide, liquid [or] solution, n.o.s.		UN2083

Fig. 2.b. Description of UN2080.

(1)	(2) Hazardous materials description and proper shipping name
Acetyl acetone peroxide , [in solution with not more than 9% by weight active oxygen. See] Organic peroxide, liquid [or] solution, n.o.s.	

(3) Hazard Class	(3A) ID	(4) Label(s) Required
	UN2080	

(5a) Exceptions	(5b) Specific Requirements	(6a) Pass. Air or Railcar	(6b) Cargo Aircraft Only

		WATER
(7a)	(7b)	(7c) Other Requirements

Fig. 2.c. Additional description for UN2080.

```
RegScan 49 Main Menu
Reg. No. 017C00-4C02A-C056B                              Updated thru Sep 01 1990

      1. Proper Shipping Name            a. Table of Contents
      2. UN/NA Number                    b. Part/Section/Subsection
      3. Table Scan Menu                 c. Word/Phrase Search
      4.                                 d. Print Section to File
      5.                                 e.
      6. Exit to DOS                     f. Aux. Services Menu

      Copyright (c) Regulation Scanning Technology Corp. 1987,1988
```

```
Please enter the Proper Shipping Name as it appears in the Table in 172.101
acetone
```

Fig. 3.a. Search by proper shipping name.

(2) Proper Shipping Name	(3) Hazard Class	(3A) ID Number
Acetone	Flammable liquid	UN1090
Acetone cyanohydrin	Poison B	UN1541
Acetone oil	Flammable liquid	UN1091
Acetonitrile	Flammable liquid	NA1648
Acetyl acetone peroxide , [in solution with not more than 9% by weight active oxygen. See] Organic peroxide, liquid [or] solution, n.o.s.		UN2080
[Acetyl acetone peroxide with more than 9% by weight active oxygen]	Forbidden	
Acetyl benzoyl peroxide , [not more than 40% in solution]. [See] Acetyl benzoyl peroxide solution, [not over 40% peroxide]		UN2081

Fig. 3.b. Resulting table for acetone.

49 CFR §173.118 Limited quantities of flammable liquids.

****

REGSCAN EDITORIAL NOTE: Changes have been made to this section according to a final rule published in the Federal Register. The original version of the changed text is presented immediately following this note. The revised version, along with effective date information, is presented in highlight immediately following the original version.

****

(a) Limited quantities of flammable liquids that do not meet the definition of another hazard class in this subchapter and for which exceptions are permitted as noted by reference to this section in §172.101 of this subchapter, are excepted from labeling (except when offered for transportation by air) and specification packaging requirements of this subchapter when packed according to the following paragraphs. In addition, shipments are not subject to Subpart F of Part 172 of this subchapter, to Part 174 of this subchapter except §174.24 and to Part 177 of this subchapter except §177.817.

Page 1 of 4

Fig. 3.c. Regulation section for flammable liquids.

(1)	(2) Hazardous materials description and proper shipping name
Acetone	

(3) Hazard Class	(3A) ID	(4) Label(s) Required
Flammable liquid	UN1090	Flammable liquid

(5a) Exceptions	(5b) Specific Requirements	(6a) Pass. Air or Railcar	(6b) Cargo Aircraft Only
173.118	173.119	1 quart(s)	10 gallon(s)

WATER		
(7a)	(7b)	(7c) Other Requirements
1,3	4	

Fig. 3.d. Hazardous material description for acetone.

```
                    RegScan 49 Table Scan Menu
                              49 CFR

         1. Scan the Hazardous Materials Table (172.101)
         2. Scan the Optional Hazardous Materials Table (172.102)
         3. Scan the CERCLA List (172.101 Appendix)
         4. Scan the RADIONUCLIDES List
         5. No Further Display, Return to Main Menu

     Copyright (c) Regulation Scanning Technology Corp. 1987,1988
```

```
Please indicate the starting scan criteria
isotopes
```

Fig. 4. a. RegScan table scan menu.

```
                    RegScan 49 Table Scan Menu
                              49 CFR

          1. Scan the Hazardous Materials Table (172.101)
          2. Scan the Optional Hazardous Materials Table (172.102)
          3. Scan the CERCLA List (172.101 Appendix)
          4. Scan the RADIONUCLIDES List
          5. No Further Display, Return to Main Menu

     Copyright (c) Regulation Scanning Technology Corp. 1987,1988
```

Select your choice by typing its number or use the UP and DOWN arrows
to highlight your selection. Then press ENTER.

Fig. 4.b. Scanning by the word isotopes.

Radionuclide	Atomic Number	RQ Curies (Tbq)
Krypton-74	36	10 (.37)
Krypton-76	36	10 (.37)
Krypton-77	36	10 (.37)
Krypton-79	36	100 (3.7)
Krypton-81	36	1000 (37)
Krypton-83m	36	1000 (37)
Krypton-85	36	1000 (37)
Krypton-85m	36	100 (3.7)
Krypton-87	36	10 (.37)
Krypton-88	36	10 (.37)

F2:Footnotes F10:Exit

Fig. 4.c. Resulting list of isotopes from the search.

Other important feature of this package is option d in the **Main Menu**, Print Section to File. By selecting "d" on the main menu, the system asks for part/section to print to file as illustrated in Fig. 5. Then the system asks for the name of the output file where the part of the section of a regulation is going to be printed. The system can also send an entire section to an output file. This feature allows the user to have a hard (paper) copy of a specific topic that can be easily incorporated in a document.

A search string can be performed through option c in the **Main Menu**. For example, a search for a Type A40 package is illustrated in Fig. 6.a. After the search has been conducted, a list of section numbers containing the string or part of the string is listed for the user (Fig 6.b). By selecting one of the section numbers, the text is immediately shown to the user (Fig. 6.c). In addition, a current chain of references is also shown to the user for additional references (Fig. 6.d) from which the user can select.

RegScan has been purchased by the EC&A through its Transportation Technologies Group as a source of regulation information. The program provides a great deal of information about 49 CFR to the user. The user interface also permits easy navigation through the regulations. Sections or parts of the regulations can be downloaded into disk files. This feature makes this program useful because it allows the regulations pertinent to transportation and packaging to be updated and downloaded on a regular basis. (RegScan updates the product every month.) The package's main limitation is that it is not suitable for work directly as an expert system for transportation and packaging problem solving. RegScan is capable of delivering information about regulations, but it is not capable of helping the user decide how to conduct a consultation for determining the type of packaging necessary for shipment of a particular hazardous material.

2.1.2 The Environmental/Safety Library

The Environmental/Safety Library** is a collection of regulatory documents on CD-ROM (compact disc-read only memory) that helps the user find regulations that apply to

**The Environmental/Safety Library is produced by Information Handling Services™, Englewood, Colorado.

```
                          RegScan 49 Main Menu
 Reg. No. 017C00-4C02A-C056B                        Updated thru Sep 01 1990

        1. Proper Shipping Name            a. Table of Contents
        2. UN/NA Number                    b. Part/Section/Subsection
        3. Table Scan Menu                 c. Word/Phrase Search
        4.                                 d. Print Section to File
        5.                                 e.
        6. Exit to DOS                     f. Aux. Services Menu

        Copyright (c) Regulation Scanning Technology Corp. 1987,1988
```

```
Please enter the Part/Section to print to file
173.403

Please name the output file (format xxxxxxxx.xxx)
173_403.reg
```

Fig. 5. Downloading a section of a regulation.

```
                        RegScan 49 Main Menu
Reg. No. 017C00-4C02A-C056B                      Updated thru Sep 01 1990

       1. Proper Shipping Name           a. Table of Contents
       2. UN/NA Number                   b. Part/Section/Subsection
       3. Table Scan Menu                c. Word/Phrase Search
       4.                                d. Print Section to File
       5.                                e.
       6. Exit to DOS                    f. Aux. Services Menu

       Copyright (c) Regulation Scanning Technology Corp. 1987,1988
```

```
Please enter the desired search string
package A
Enter the number of occurences you wish to locate or press ENTER for all
all
Enter the starting section number for the search or press ENTER for first

Enter the ending section number for the search or press ENTER for last
```

Fig. 6.a. Search for Type A package.

```
                         RegScan 49
                         Search Menu

        1. Display Text for Current List
        2. Show Next Set of Section Numbers
        3. Show First Set of Section Numbers
        4. No Further Display, Return to Main Menu

  Copyright (c) Regulation Scanning Technology Corp. 1987,1988
```

```
Search string:
package A
Section Numbers Containing String:
49 171.11(c)
This is set 1  of 1  sets returned
```

Fig. 6.b. Section numbers containing string for Type A package.

49 CFR §171.11 Use of ICAO Technical Instructions.

****
REGSCAN EDITORIAL NOTE: Changes have been made to this section according to a final rule published in the Federal Register. The original version of the changed text is presented immediately following this note. The revised version, along with effective date information, is presented in highlight immediately following the original version.
****

Notwithstanding the requirements of Parts 172 and 173 of this subchapter, a hazardous material may be transported by aircraft, and by motor vehicle either before or after being transported by aircraft, in accordance with the ICAO Technical Instructions if the hazardous material:

(a) Is packaged, marked, labeled, classified, described and certified on a shipping paper and otherwise in a condition for shipment as required by the ICAO Technical Instructions;

(b) Is within the quantity limits prescribed for transportation by either passenger-carrying or cargo aircraft, as appropriate, as specified in the ICAO Technical Instructions;

(c) Is not a forbidden material or package according to §173.21 or Column (3) of the Table to §172.101 of this subchapter; and,

Page 1 of 7

Fig. 6.c. Text for the search.

Current chain of references:

Return to menu

49171.11(c)

49172.203(d)(1)(vii)

49172.402(b)

49172.402(c)

49173.3(c)(2)

Press F10 to return to Main Menu or use UP, DOWN, LEFT, or RIGHT arrows keys to highlight section then press ENTER.

Fig. 6.d. List of references from which the user can choose.

the specific situation. The system accesses U.S. Environmental Protection Agency (EPA) and Occupational Safety and Health Administration (OSHA) Public Laws, 40 CFR, 20 CFR 1900-End, and 49 CFR 100-199. It accesses this information by part/section number, subject, date of publication, or key word. It also has the capability of showing figures.

The Environmental/Safety Library on CD-ROM is being introduced by IHS Regulatory Products (a subsidiary of IHS, Inc.). The disk contains a collection of regulations, including EPA, OSHA, and Department of Transportation (DOT) regulations and laws. The Environmental/Safety Library contains the full text of the following documents:

- EPA and OSHA Public Laws [including the Clean Air Act, the Clean Water Act, the Resource Conservation and Recovery Act (RCRA), the Superfund Amendments Reauthorization Act (SARA), the Toxic Substances Control Act (TSCA), CERCLA, OSHA Act of 1970, etc.],

- 40 CFR (complete EPA CFR data base including RCRA, SARA, etc.),

- 29 CFR 1900- End [complete OSHA CFR, including the Construction Regulations (part 1926)],

- 49 CFR 100-199 (DOT CFR on the Transportation of Hazardous Materials), and

- Federal Register Final & Proposed Rules pertaining to the OSHA, EPA, and DOT regulations cited above.

Figure 7 illustrates the main menu of the program while performing one search on the demo provided by IHS Regulatory Products. Figure 8 illustrates a word search feature of the program. Figures 9 and 10 illustrate a real search for the word "disposal."

```
ENVIRONMENTAL/SAFETY LIBRARY
F1     F2                                                             F10
Help   Searc              Demonstration Main Menu              ls     Quit

               A)    Overview of Search Screen
               B)    Select Database
        En     C)    Browse Table of Contents
               D)    View and Search from Word List Dictionary
               E)    Examine and Change Search Options
               F)    Search from Main Search Screen
        Ti     G)    View Result List
               H)    View Document
               I)    Perform Side Search
               J)    Bookmark List
      Subj     K)    Notepad Facility
               L)    Context-Sensitive HELP Facility
               M)    Images
        Pa
               Q)    Exit demonstration

                        Press LETTER of your choice
      Mult
      Use asterisk (*) as wildcard character.  Press F2 to Search.
```

Fig. 7. Main menu of environmental/safety library.

```
ENVIRONMENTAL/SAFETY LIBRARY
F1     F2        F3    F4          F5     F6 Srch      F8 Srch to    F9        F10
Help   Search    List  Select DB   TOC       Options      Document   Tools     Quit

      ─────────────────────── Search by Keywords ────────────────────────

         Enter word(s) ____________________________________________
                    or ____________________________________________
                but not ____________________________________________
                    or
         Title word(s) ____________________________________________

      ═══════ or ════════════ IHS  Subject Index
                                                   ┌──────────────────────────────┐
       Subject word(s) __________________________  │ You are in the Search        │
      ═══════ or ══════════════ Part/Section No =  │ Screen.  Press <DOWN>        │
                                                   │ to continue.                 │
                                                   └──────────────────────────────┘
         Part/Sect No: ____________________________________________
      ══════ and ═════════════ Part/Section Date ══════════════════════
           Start Date: ______________      End Date: ______________
      ─────────────────────────────────────────────────────────────────
       Multiple words on a line are AND'd. Enclose phrases in "  ".
       Use asterisk (*) as wildcard character.  Press F2 to Search.
```

Fig. 8. Search mechanism.

```
ENVIRONMENTAL/SAFETY LIBRARY
F1     F2       F3    F4          F5     F6 Srch     F8 Srch to    F9       F10
Help   Search   List  Select DB   TOC       Options     Document   Tools    Quit

     ─────────────────────── Search by Keywords  │ This is the number of
                                                 │ documents in the database
       Enter word(s) disposal────────────────────│ that contain the word
               or    ────────────────────────────│ 'disposal'.
          but not    ────────────────────────────│ Press  <F8> to go to the
         ────────────────────────────────────────│ first document.
         No. of Documents found = 3.
         Press F2 to go to Results List.
     ══  Press F8 to go to First Document.          ══════════════════

     Subject word(s) ────────────────────────────────────────────
     ═══════ or ══════════════ Part/Section No ═════════════════════

       Part/Sect No: ────────────────────────────────────────────
     ══════ and ═════════════ Part/Section Date ═══════════════════
         Start Date: ──────────────       End Date: ──────────────
     ─────────────────────────────────────────────────────────────────
     Multiple words on a line are AND'd. Enclose phrases in "  ".
     Use asterisk (*) as wildcard character.  Press F2 to Search.
```

Fig. 9. Image search.

```
ENVIRONMENTAL/SAFETY LIBRARY
F1     F2         F3    F4          F5     F6 Srch      F8 Srch to    F9       F10
Help   Search     List  Select DB   TOC       Options      Document   Tools    Quit

     ┌────────────────── Search by Keywords ───────────────────────┐
     │                                                              │
     │   Enter word(s) disposal____________________________________ │
     │           or    ____________________________________________ │
     │      but not    ____________________________________________ │
     │           or                                                 │
     │   Title word(s) ____________________________________________ │
     │                                          ┌─────────────────────────────┐
     │══════ or ══════════ IHS  Subject Index   │ Images may be viewed        │
     │                                          │ from within the document    │
     │ Subject word(s) _________________________│ in a Searched or Browsed    │
     │══════ or ════════════ Part/Section No ═  │ Screen.                     │
     │                                          │ You will now go to a        │
     │   Part/Sect No: _________________________│ document that has images    │
     │═════ and ═══════════ Part/Section Date   │ and will VIEW them.         │
     │     Start Date: ______________     End Da│                             │
     │──────────────────────────────────────────│ ENTER the word 'disposal'   │
     │ Multiple words on a line are AND'd. Enclos│ Press <CR> to start the    │
     │ Use asterisk (*) as wildcard character.  P│ search.                    │
     └──────────────────────────────────────────┴─────────────────────────────┘
```

Fig. 10. Stage two of an image search.

The Environmental/Safety Library was demonstrated by the vendor's representative. The program provides a massive amount of information to the user. The user interface allows easy navigation through the regulations. Sections or parts of the regulations can be downloaded into disk files. This feature makes the program useful because it allows the regulations pertinent to transportation and packaging to be downloaded on a regular basis. (IHS updates the product every two months.) According to IHS's representative, this information could be distributed within the U.S. Department of Energy (DOE) without additional costs. The limitations to this package include the additional requirement of a CD-ROM drive connected to the IBM-PC compatible, a rather long response time for each data base query, and an inability to work as an expert system for transportation and packaging problem solving. The Environmental/Safety Library is capable of delivering a great deal of information about regulations, but it is not capable of assisting the user in the type of hazardous material packaging required.

2.1.3 Conclusions Concerning Availability

RegScan and the Environmental/Safety Library are two useful commercial computer programs for regulation scanning. However, neither program can be used as expert systems since they do not contain an inferential algorithm that takes the user through the decision-making process for transporting radioactive and hazardous waste. However, both programs have the capability of delivering periodically updated versions of the regulations that may eventually be consulted in the expert system. RegScan is analyzing the possibility of licensing the use of the updated regulations for a wider distribution within an expert system. In principle, the developer is interested in discussing a joint development agreement with DOE (a CRADA has been mentioned as a possibility). It is believed that such an agreement will be mutually beneficial. Regulation Scanning would be interested in negotiating a broader licensing arrangement to permit regular access to updated RegScan data for a wide base of users. Similarly, the Environmental/Safety Library representative stated that the use of updated information from their product could be distributed within DOE without additional cost. Corroboration of this statement is being obtained from the company's management.

2.2 ANALYSIS OF COMPUTER TOOLS TO BE APPLIED IN THE EXPERT SYSTEM DEVELOPMENT

There are several tools that are specifically adaptable for developing expert systems. Design requirements for the expert system include easy access to data base information, the capability to execute complex calculations quickly, graphics display facilities, and, most importantly, a user-friendly interface. An expert system may be built using a shell, which is a program designed specifically for expert system development, or an AI language.

The expert shell is a computer program that provides a fixed set of features for developing expert systems. It has the advantage of having already been developed, tested, and debugged. The expert shell can load and consult different knowledge bases and apply the same rules of inference to solve a variety of problems. With an expert shell, however, the developer is largely constrained by the features that were originally designed into it. Another disadvantage associated with using a shell is related to the distribution procedures. In order to distribute the application, each user has to purchase a run-time version in order to use it.

On the other hand, directly building the expert system using an AI programming language gives the developer total flexibility in providing features that are specific to the needs of a problem. In addition, the application may be distributed as an executable version for which no additional fee will be paid. On the negative side, however, a considerable amount of code would need to be developed.

One of the objectives of this project involves analyzing at least two different AI languages—PDC Prolog and Smalltalk. Additionally, the analysis of two other computer packages— Borland C++ and GUIDE—will also be considered. PDC Prolog is a computer language that is primarily based in predicate logic that uses **IF**, **THEN** situations to express knowledge (Prolog stands for programming in logic.). This language is available for IBM-PC compatible machines. Smalltalk is a computer language that uses object-oriented techniques for programming and is available for IBM-PC compatible machines that work under Microsoft® Windows 3.0.[3] Borland C++ is a programming language available for IBM-PC compatible machines that works under DOS and Windows 3.0. GUIDE is a commercial shell that works under Windows 3.0 and offers users several interesting features, such as video image handling and hypertext documents.

2.2.1 PDC Prolog

PDC Prolog*** is produced by Prolog Development Center. PDC Prolog is a declarative language, which means that, given the necessary facts and rules, PDC Prolog will use deductive reasoning to solve programming problems. This type of language is in contrast to traditional computer languages, such as BASIC and Pascal, that are procedural languages. In procedural languages, the programmer must provide step-by-step instructions that tell the computer exactly how to solve a given problem. In other words, the programmer must know how to solve a given problem before the computer can do it. In comparison, the PDC Prolog programmer only needs to supply a description of the problem and the ground rules for solving it. From there, the Prolog system is left to determine how to find a solution.

PDC Prolog has several features that make it very attractive for expert system development. The rules can be built using conventional **IF, THEN** clauses. In order to prove a particular hypothesis, the system will backtrack to its original conditions, prove them, and then make the hypothesis true; thus, activating the actions that evolve from the hypothesis. One example of such a rule could be:

```
packageB :-        material = special_form and
              total_activity > A1_limits.
```

which means:

IF material = special form and total activity is greater than A1 limits
THEN
package type is B.

***PDC Prolog is produced by Prolog Development Center, Atlanta, Georgia.

Other features of Prolog include simple data base handling commands. Data bases written in PDC Prolog can be stored in expanded memory, regular memory, or hard disk. The last option of keeping the data base in hard disk may represent one of the outstanding features of PDC Prolog, although this type of storage does delay the retrieval of information for a few moments. Translating the advantage of this feature into practical terms, the amount of information that may be held in the data base is limited only by the size of available disk space.

PDC Prolog can also access text files that may hold massive amounts of information. These text files can also reside in the hard disk. A word or group-of-words search routine can be easily set up to configure the text files.

PDC Prolog works with a graphical environment and can handle graphics produced by Borland Graphics Interface (BGI) commands or bit images with GIM format.

Presently, one disadvantage of PDC Prolog is the 640K working memory ceiling, which is typical for a DOS application. However, the next version of PDC Prolog for Windows 3.0 has been extensively beta tested and will be on the market shortly. This new version promises to work with the maximum memory available in the IBM compatible computers (typically 2-8 MB). Another current disadvantage of PDC Prolog is the type of graphics that can be directly accessed by the program. It can access bit images in GIM format, which is not extensively used by other software packages or hardware systems (such as a scanner). However, a program that takes images directly from a scanner to be later translated into Prolog commands, has been written as part of this project. The Windows 3.0 version will allow manipulation of bitmap graphic files that are commonly used by other systems.

2.2.2 SmallTalk

OOP is one of the central programming paradigms to emerge in this decade. The scope of its influence can be seen in the introduction of objects into numerous programming languages such as Loops, Object-Pascal, C++, and Prolog, among others.

Smalltalk/V is one of the most extensively used object-oriented languages. This language offers a completely new environment for software development.

Smalltalk/V Windows (developed by Digitalk, Inc.) provides pure OOP, a revolutionary approach to data abstraction, and a new dimension to organizing the elements of a software system.[3] The combination of the Windows graphical interface and the programming environment of Smalltalk/V yields a powerful software development tool.

Applications developed in Smalltalk are created in terms of objects and methods (subroutines) specific to those objects. The basic environment contains objects and methods that make windows and icons easy to create. Because of these features, the development of an operator interface is very straightforward. Another benefit of using the Smalltalk environment is the presence of all source code. In many instances, a source code can be reused when new applications are being developed. Flexibility is provided by the presence of a source code since the base system can be enhanced by including slight modifications of the existing code. Inheritance is another feature of Smalltalk that reduces development time. The objects present are in a hierarchy of parent-child relationships, which means that when an object is created as a subclass (child) of an existing object (parent), the child inherits all of the parent's methods. However, if a method is created at the child level with the same name as a parent method, that method will override the method of the parent. This feature eliminates the need for duplication of code.

There are many advantages for using Smalltalk/V. Installing and modifying small modules of code can be done without completing the lengthy process of compiling and linking the entire program; bits and pieces of the program can be experimented with before the entire program is developed, and debugging the program is smoother and faster because errors are easier to detect within smaller modules of code. In addition, Smalltalk/V allows the user to reuse as many lines of prewritten code as possible and modify them for several different types of programs. Smalltalk/V allows the access of text files containing massive amounts of information, data bases written in Smalltalk/V, data bases written in Dbase III, and graphic bitmap files.

However, there are also some disadvantages of using Smalltalk/V. The disadvantages reside in the preparation of a run-time version for wide distribution. The user must have Smalltalk/V on his/her machine to run the application, and learning to program in Smalltalk/V is time consuming, therefore when future changes in the application are required, only a skillful programmer in Smalltalk/V can make them in a short period of time.

2.2.3 GUIDE

GUIDE**** is a multimedia software created by OWL International that is implemented in Windows 3.0. GUIDE software allows the user to navigate through large amounts of material to locate answers quickly. In interacting with information through associative links, users can follow their trains of thought and view any level of detail they desire. By activating a pointing device (such as a mouse) on specific objects, users access cross-referenced materials, footnotes, reports, articles, graphs, full-color illustrations, spreadsheets, video and audio messages, etc.

GUIDE helps the developer create and maintain interrelated documents. A built-in programming language, LOGiiX, gives the user control of all GUIDE functions, thus allowing for the development of "smarter" documents that will simplify reader interaction. Features include: comprehensive document formatting; graphical user interface; flexible link structures; text, graphics, video, and sound versatility; and search and object management tools.

The applications made with GUIDE look extremely professional—especially those packages designed for training purposes. However, the construction of an expert system may require additional programming skills from the developer. The distribution of the final application will require purchasing the *GUIDE Reader*, the price of which varies depending upon the number of copies desired. If the application includes video images, the system owned by the final user must have similar video and graphic interface hardware as with the computer where the application was developed.

****GUIDE is produced by OWL International, Bellevue, Washington.

2.2.4 C++

Initial efforts were made to put the reasoning mechanism in C++. The use of this language did not present any advantage for the development of the expert system prototype. Therefore, no further attempts with C++ were made.

3. EXPERT KNOWLEDGE ACQUISITION

Ordinary computer programming is usually characterized as data manipulation. AI programming, on the other hand, can be thought of as knowledge manipulation. Therefore, the AI programmer must deal with the issue of knowledge representation, because the knowledge to be manipulated must be represented in some manner. The developer and the expert(s) must be in close interaction in order to converge the knowledge base into a manageable set of representation items.

3.1 LOGIC DIAGRAM DEVELOPMENT

Work done by Tarnuzzer[2] demonstrates that knowledge about transportation packaging for radioactive materials can be represented using a logic diagram. The logic diagram for radioactive materials transportation packaging is represented in Fig. 11.

Figure 11 illustrates the five different stages necessary to determine the type of package required for transporting a given radioactive material. Stage 1 of the logic diagram simply represents the data input required from the user. At this point, the user must answer if the material conforms with the definition of radioactive material. In addition, the user needs to supply information about the isotopes that comprise the radioactive material [i.e., how much activity per isotope, the form of the material (special or normal), and the physical state of the material (solid, liquid, or gas)]. Along with this information, stage 1 of the diagram contains enough data for the program to calculate the

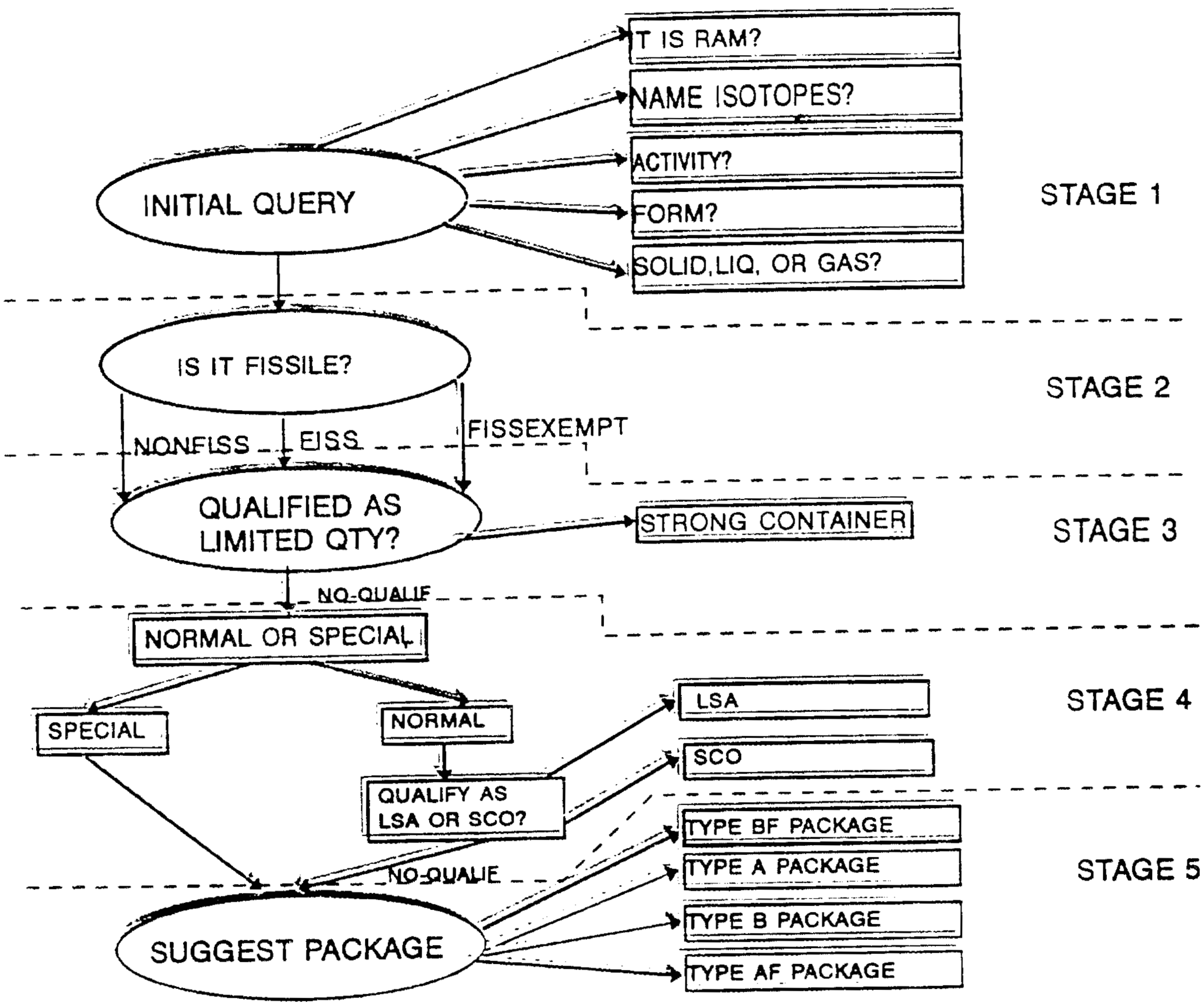

Fig. 11. Logic diagram for radioactive materials transportation.

fraction of A_1 (if material is special form) or the fraction of A2 (if material is normal form). The fraction of A1 or A2 is calculated using the following formula:

Fraction(A1 or A2) = SUM [(Activity Isotope i)/(A1 or A2 isotope i)].

Decisions will be based on the value of this fraction and will determine the path for the following stages of the logic diagram.

Stage 2 of the logic diagram determines whether the material to be transported is fissile, nonfissile, or fissile exempted. Figure 12 illustrates stage 2 of the logic diagram. The double boxes on the logic diagram represent decision points that the system can do by itself. Hence, enough information has been provided by the user or enough facts have been matched with the rule system to reach a decision. Simple boxes represent points in which the user is required to enter some kind of information. Next, a portion of stage 2 of the logic diagram will be analyzed. The analysis is analogous for the other portions of stage 2.

At the beginning of stage 2, the program is capable of determining if the fissile isotopes have been identified explicitly (e.g., "no selection of uranium enriched"). Once the system determines that fissiles radionuclides have been identified, it performs computations to calculate the mass in grams (g) of the respective fissiles. If the mass is less than 15 g, the material is excepted from the fissile material requirements. If there are more than 15 g of fissiles, the system asks the user if the package contains uranium enriched in ^{235}U to a maximum of 1% by mass and mixed with a total plutonium and ^{233}U content of up to 1% of mass of ^{235}U. A positive answer from the user will prompt the system to ask whether the fissile material is distributed homogeneously throughout the package contents rather than from a lattice arrangement within a package. An affirmative answer will lead the system to determine that the material is excepted from fissile material requirements. A negative answer with the knowledge that the material is, for example, liquid, prompts the system to ask whether the package contains ≤ 5 g of fissile radionuclides in any 10-L volume, and whether the limit will be maintained throughout

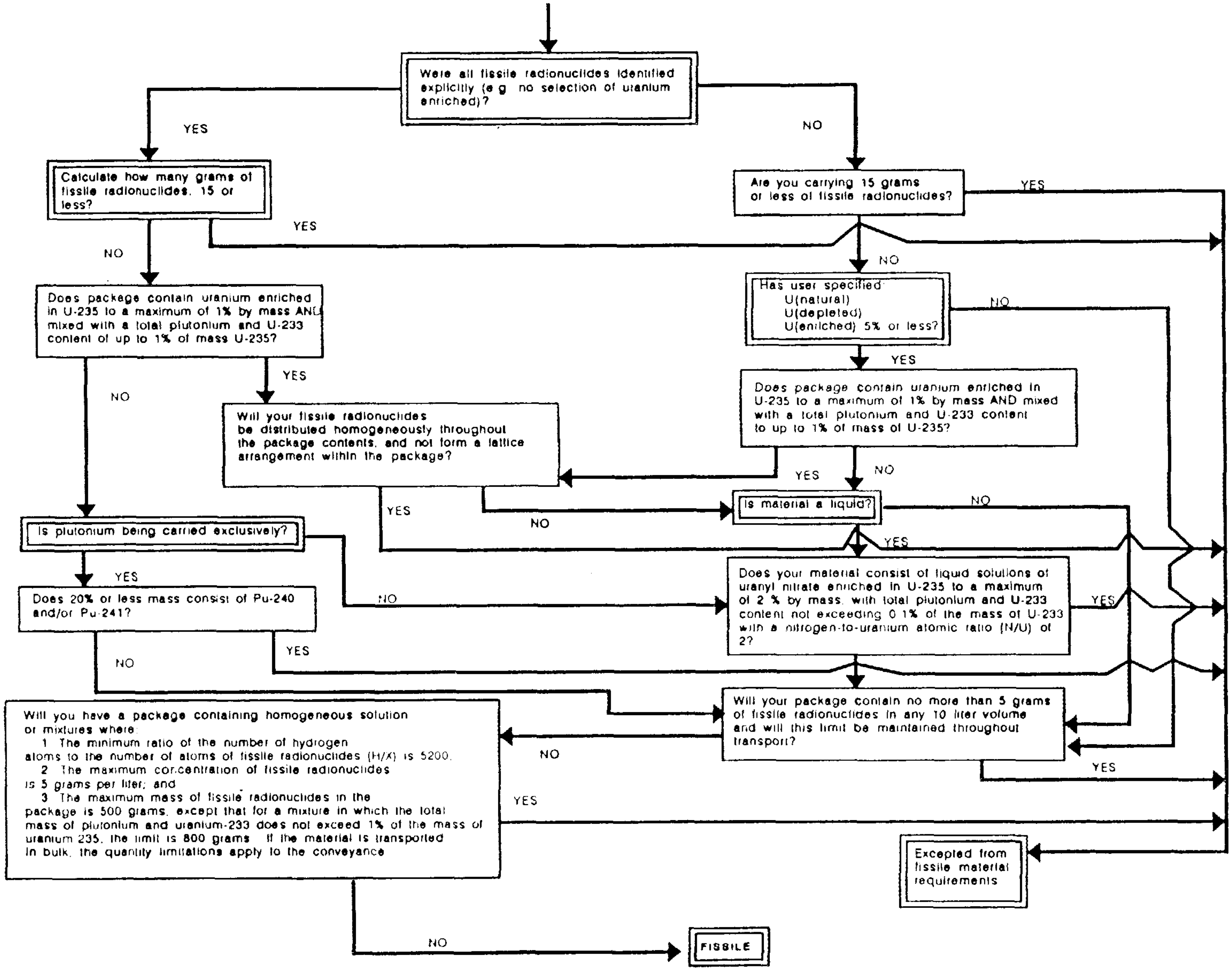

Fig. 12. Stage 2 of the logic diagram for radioactive materials transportation packaging.

transport. A positive answer implies an exception of fissile material requirements. A negative answer will confront the user with characteristics of the package containing homogeneous solutions or mixtures. Also, a positive answer to this last question implies an exception from fissile material requirements. A negative answer implies that the material is fissile and has to conform with fissile material requirements.

Stage 3 of the logic diagram is illustrated in Fig. 13. Ultimately, the system determines if the material qualifies as limited quantity. Again, only a portion of this stage will be analyzed. Once the system has checked that the material is less than Type A quantity, it determines whether the sole radioactive contents are natural uranium, depleted uranium, or natural thorium. A positive answer to this last question prompts the system to ask if the material consists solely of manufactured articles, in which the outer surface of the uranium or thorium is enclosed in an inactive sheath made of metal or other durable protective material. A positive answer to this question causes the system to determine that the material can be transported as limited quantity and suggest shipment per 49 CFR 173.424. A negative answer causes the system to ask if the user is shipping instruments or articles. If the user is shipping articles, the system looks for the article(s) in Table 7, 49 CFR and determines whether the material meets the limits shown. If the limits are met, the material can be shipped per 49 CFR 173.422. If the limits are not met, then the system determines that the material has not qualified as limited quantity.

Stage 4 of the logic diagram is illustrated in Fig. 14. Stage 4 determines, for those materials that have not qualified as limited quantity and have been declared as normal, whether the material can be shipped as Low Specific Activity (LSA) or Surface Contaminated Object (SCO). Once the system has determined that the total activity is less than 2-A_2, the user is in a position to answer whether the material could be shipped as LSA or SCO. The portion of the diagram that determines if the material can be shipped as LSA is illustrated in Fig. 15. The portion of the logic diagram that determines if the material can be shipped as SCO is illustrated in Fig. 16. Next, the analysis of determining if the material can be transported as LSA will be described. The analysis for SCO is analogous.

The logic diagram illustrated in Fig. 15 begins by determining if the material is a gas. The system searches its memory to see if the material in question is a gas or not. If

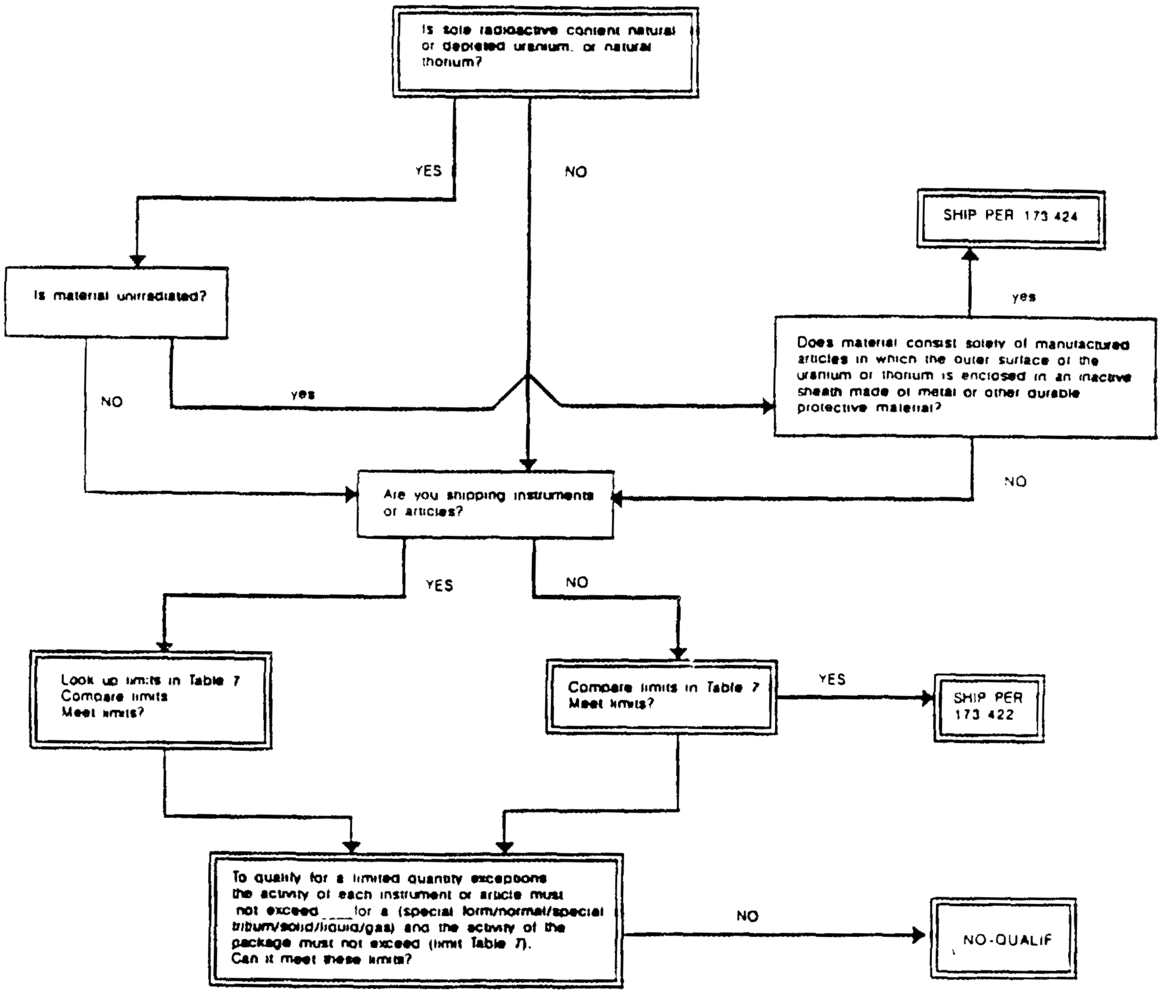

Fig. 13. Stage 3 of the logic diagram for radioactive materials packaging.

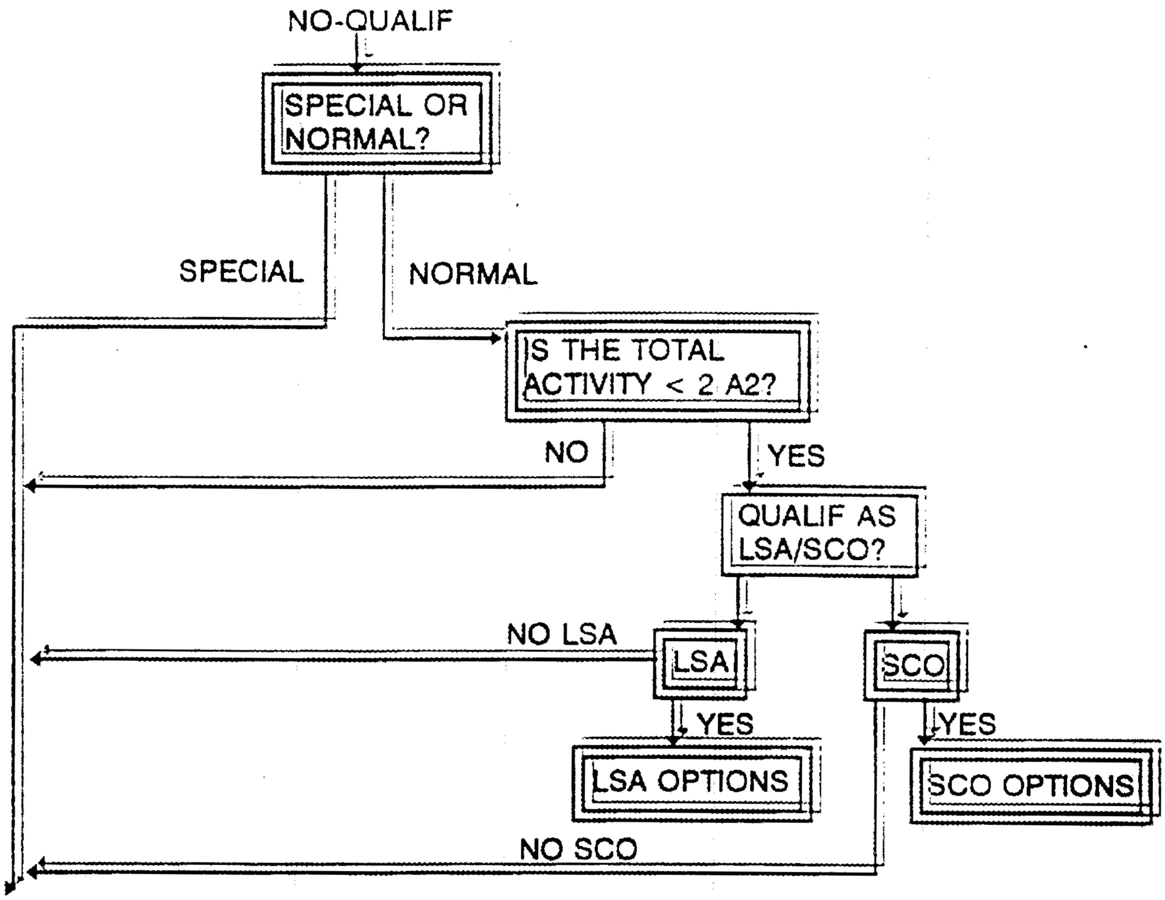

Fig. 14. Stage 4 of the logic diagram for radioactive materials packaging.

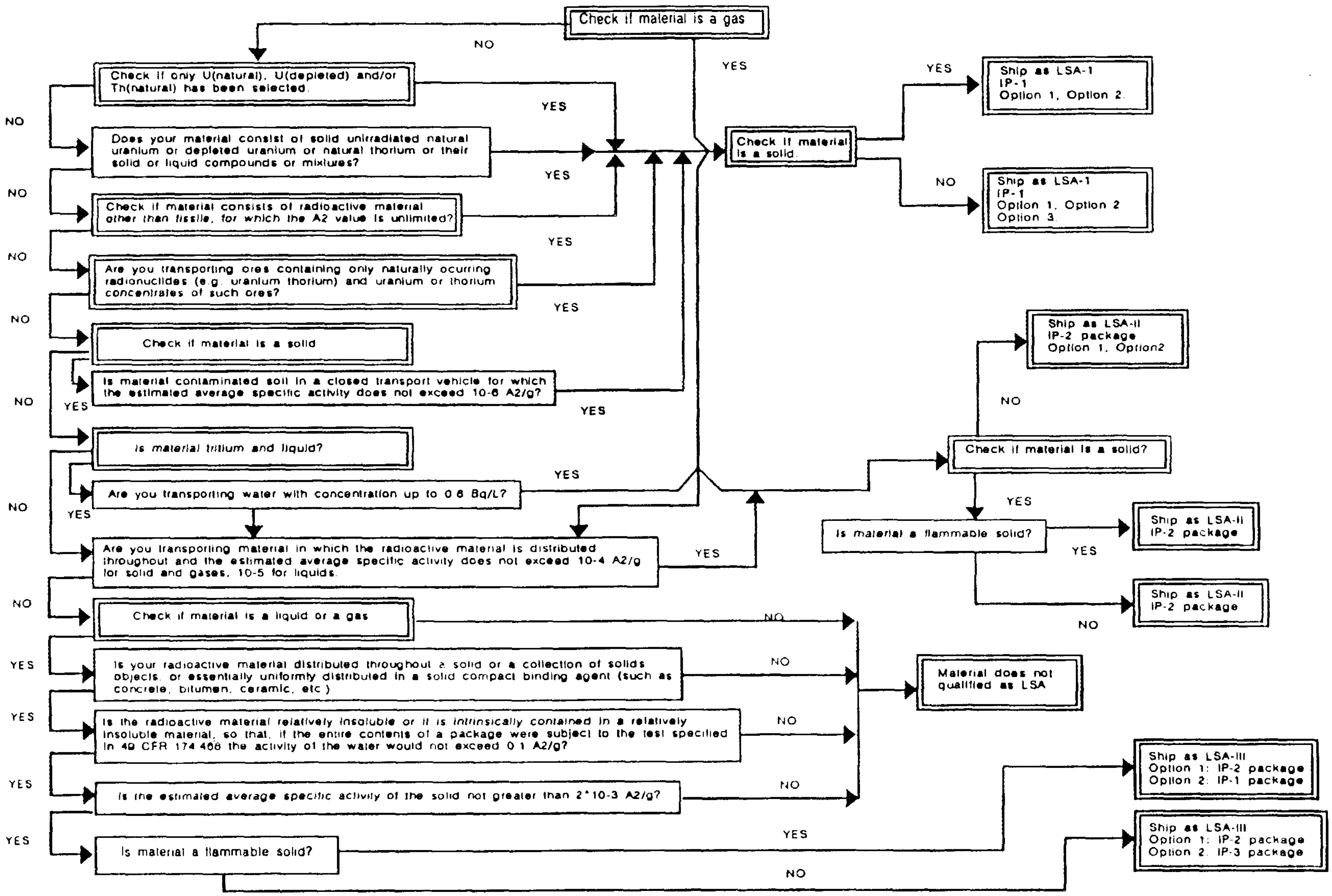

Fig. 15. Portion of logic diagram for low specific activity qualification.

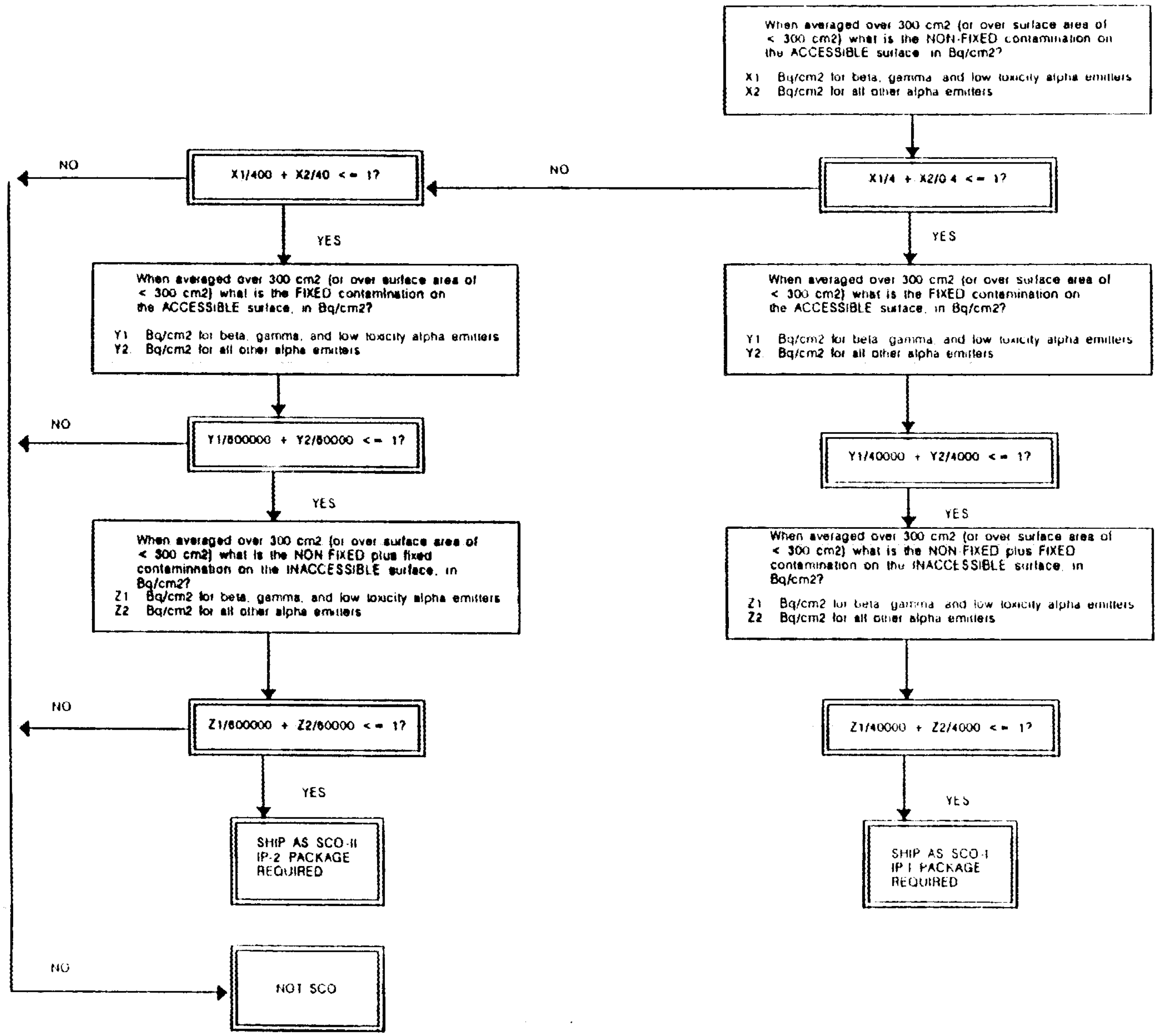

Fig. 16. Portion of logic diagram for surface contaminated object qualification.

the system finds a positive answer, it asks the user whether the radioactive material is distributed throughout the material to be transported, and whether the estimated average specific activity does not exceed 10^{-4} A_2/g for gases. A positive answer from the user causes the system to query itself again as to whether the material is a solid. Since it is a gas, the system determines that the material should be shipped as LSA-II with IP-2 package.

If the material is not a gas, the system checks its memory to determine if the isotopes are only U(natural), U(depleted), and or Th(natural). If the answer is yes, then it checks whether the material is a solid or a liquid. If the material is a solid, then the system suggests that the material should be shipped as LSA-2 in packages IP-1, options 1 and 2. If the material is a liquid, then the system suggests the material should be shipped as LSA-I in packages IP-1 with options 1, 2, and 3.

If the material is not only U(natural), U(depleted), or Th(natural); the system asks the user if the material consists of solid unirradiated natural uranium, depleted uranium, or natural thorium (or their solid or liquid compounds, or a mixture thereof). A positive answer by the user with a liquid material prompts the system to suggest that the material be shipped as LSA-1 in packages IP-1, options 1, 2 and 3. If the material is solid, then the system suggests to ship the material as LSA-2 in packages IP-1, options 1 and 2.

If the material does not consist of an unirradiated solid, the system checks its memory to determine whether the material is fissile. If the material is fissile, the system checks whether the material consists of ores containing only naturally occurring radionuclides and uranium, or thorium concentrates of such ores. If the answer is negative and the material is a solid, the system asks the user if the material is contaminated soil in a closed transport vehicle for which the estimated average specific activity does not exceed 10^{-6} A_2/g. If the answer is no, the system checks whether the material is distributed throughout and the estimated average specific activity does not exceed 10^{-4} A_2/g for solids. A negative answer from the user causes the system to respond that the material does not qualify as LSA. The procedure to analyze other combinations of questions and answers is analogous to what has been mentioned above.

3.2 FEATURES REQUIRED FROM THE EXPERT SYSTEM PROTOTYPE TO SATISFY LOGIC DIAGRAM

The logic diagram will be translated into a computer program that will be accessed by the user. The user may or may not have the knowledge necessary to run every portion of the program in order to obtain a satisfactory answer. The first concern of the knowledge engineer is to develop an interface between the user and all the specific program(s) that is transparent regarding the manipulations that are required to go from one set of input data, to calculation programs, to decision maker programs, etc. This feature is normally referred to as the "friendliness of the interfacing." One of the main characteristics of friendliness is easy access to the main program. Thus, a straightforward system of a question-answer relationship between the computer and the user is highly recommendable. It is advisable to put the questions in one area of the computer screen and the answers to be given by the user in a separate area within the screen. Normally, this is obtained through the use of different windows.

The type of knowledge normally involved in a logic diagram is well represented by a rule-based system that uses symbolic logic—particularly first-order predicate logic, which is widely used in AI applications today. The AI language, Prolog, is built on predicate logic, which means that once the knowledge is coded in logical symbols, the methods of reasoning with that knowledge are clearly defined and understood. Every decision point in the logic diagram can be represented by an IF, THEN rule. Consequently, the expert system has to be implemented in an environment that handles rule development.

Several types of information sources are suggested by the logic diagram. Behind every question asked of the user, there is a set of regulatory requirement(s) that will drive the answer given by the user to a specific decision point. Thus, it is obvious that the user will have to access the regulations in some cases before an answer is given to the question. This feature provides the less knowledgeable user enough information to give an appropriate response. It is a well-known fact that regulations normally refer to other regulations or parts of regulations, which in turn may refer to other regulations and so on. In such complex cases, the expert system not only needs to access regulations, but it must have the ability to browse through them. Therefore, a mechanism must be in place to access the other referenced regulations.

Other types of information that the system will access is contained in data bases. One of these data bases contains the isotopes of Table of A_1 and A_2 values for Radionuclides from 49 CFR 173.435. Every record of this data base contains the isotope name, the value of the specific activity in Ci/g, the value of A_1 and the value of A_2. Consequently, it is no surprise that one of the important features of the system involves accessing data bases.

Once a decision about a package type has been reached, additional information will be provided to the user. This information will include a graphical description of the specific types of packages. The program has to be able to retrieve explanatory graphics of packages that may reside within a data base.

Additional explanations about regulations, interpretation of regulations, or any other aspect of the decision mechanism to determine types of packages may be available in a form of video images. It would be most appropriate if this visual information could be accessed by the user, since audiovisual help is normally a very efficient way of delivering messages or knowledge. The final package would then be required to have audiovisual features.

4. PROTOTYPE DEVELOPMENT

The feasibility of developing a full expert system for transportation and packaging of hazardous and radioactive materials has been initiated within the framework of three subtasks: (1) analysis of commercial programs related to regulation scanning, (2) analysis of computer languages to develop the expert system, and (3) development of expert system prototypes. The strategy for the latter subtask is to: a) develop modules to capture the knowledge of different areas of transportation and packaging and b) analyze the feasibility of appending these different modules in one final package. The individual modules contemplate one prototype for transporting and packaging radioactive material and another for transporting hazardous chemical materials. In the event that it is not feasible to link these two systems, the modules can always be used as stand-alone tools, or linked as a single tool with some restrictions in their applicability. The subsequent sections of this report will address the development of the module to determine transportation packaging requirements for radioactive materials.

The three previously analyzed computer languages (Smalltalk, Prolog, and GUIDE) show great development potential for the expert system. Perhaps, the less adaptable language, as far as meeting the requirements and features mandated by the logic diagram, is Smalltalk. The language is powerful, but the incorporation of new features is very slow due to the abstraction of the language. In addition, the learning curve is extremely slow. To compound the disadvantages of Smalltalk, the manuals explaining the syntax of the language are poorly written. In essence, the computer tools used to develop the expert system prototype are the AI language, Prolog, and the hypermedia environment, GUIDE.

The analysis of the commercially available software—RegScan and Environmental/Safety Library—indicated that both packages, although very useful for navigating the pertinent regulations, are not particularly suitable for the determination of the type of packages required for hazardous and radioactive material transportation. However, the regulations can be downloaded from these software systems and used as the source data base for the regulations that will be accessed by the expert system.

4.1 CHOOSING THE PROTOTYPE DEVELOPMENT ENVIRONMENT

From the analysis of the implementation of the different features required for the expert system prototype, it was concluded that the development efforts should be directed toward two versions of the prototype. First, a simple DOS version that could be run on any computer (thus accelerating the review process) would be desirable. Second, a more sophisticated Windows Hypermedia version that includes every required feature must be built as a proof-of-concept that the entire software system can be built.

The DOS model essentially serves two purposes: 1) the logic reasoning is easier to analyze and review with a simpler model, and 2) the same reasoning mechanism can be used in the more sophisticated model. The only difference between the two versions is the types of features that each model can access.

4.1.1 Developing the DOS Version Prototype

The DOS prototype was developed using PDC Prolog, because it allows more flexibility with the different features incorporated in it. The distribution of the prototype, for revision purposes, may be simpler than with the other programs. The general strategy to develop this module has been to use the same logic diagram of the decision making process for transporting and packaging radioactive material (RAM) and, second, to place the expertise in Prolog code.

The general sequences of subprograms that comprise the prototype is illustrated in Fig. 17. The main program has been linked to other external subprograms. As shown in Fig. 17, the user must interact with the program and answer some basic questions.

4.1.2 Developing a More Sophisticated Windows Hypermedia Version Prototype

A more sophisticated version using hypermedia features are used to develop the prototype. Hypermedia technology usually works as an interactive software system that gives PC users the ability to organize, manage, and present information in a number of formats—text, graphics, sounds, and full-motion video. The package that has been selected for implementing the prototype is GUIDE from OWL International. The GUIDE hypermedia system is designed around an object-oriented information model that represents information as a series of linked "objects" and manages the relationships between them. Every component in a GUIDE document, whether it's a single word, phrase, paragraph or graphic, is represented as an object.

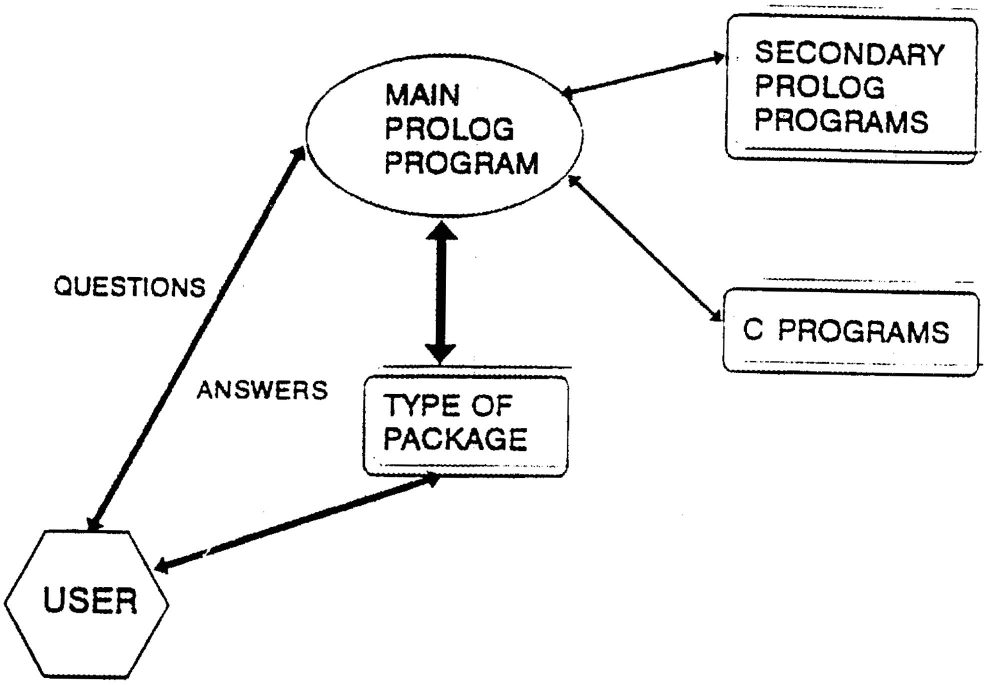

Fig. 17. DOS version prototype architecture.

GUIDE provides a variety of object types that perform different functions, including buttons, which enable users to move quickly and easily between related topics in a GUIDE document. Using a mouse, readers click on buttons to display linked objects.

GUIDE may use LOGiiX, an embedded programming language that allows the user to perform complex information manipulations. LOGiiX enables the user to add new capabilities that are not available in GUIDE's standard feature set to the hypermedia application. One of these new capabilities would include the possibility of adding the reasoning mechanism developed in the DOS version. A rule-based system that permits the development of expert systems is not included in GUIDE. However, LOGiiX provides commands that can access rule reasoning mechanisms at any point during the consultation. The reasoning mechanisms and user interaction will be developed in Prolog and accessed through GUIDE LOGiiX commands.

4.2 ANALYSIS AND IMPLEMENTATION OF REQUIRED FEATURES

The general requirements for the expert system prototype in terms of features can be summarized as follows:

- access and browse mechanisms for text file regulations,
- accessible graphic files of package descriptions,
- rule-based mechanisms,
- accessible audiovisual information,
- user-friendly interface, and
- accessible data bases.

4.2.1 Implementation of Access and Browse Mechanism for Text Files

Determination of the type of package required to transport radioactive materials is relied heavily upon knowledge of 49 CFR parts 171 to 177. In order to answer some of the questions asked by the computer, the user may need to browse through a particular portion of a regulation. Access to this type of information is usually provided by two basic search systems—menu driven and hypertext, depending on whether the DOS or Windows Multimedia version has been implemented. The menu and hypertext mechanisms work with text files, which means that the regulations have to be downloaded in an ASCII format from their original electronic source. Minor changes are required for these files to be workable in one of the two search systems.

The regulations or portions of regulations required by the prototype are downloaded from RegScan and used as the source data base for the regulations. RegScan downloads the regulations in ASCII format.

DOS version. The DOS version uses a menu-driven mechanism. This is the most common mechanism for accessing text files. In general, a menu is presented to the user from which an item can be selected. Every item represents a part or a section of a regulation. There will be cases where one regulation section refers to another regulation section, in which case another menu would have the referred regulations as choice items. To obtain the desired effect, a pull-down menu having at least three submenus will provide the desired feature. The first submenu (Regulations) gives the choices for the main set of regulations, the second submenu (Related Regs) gives the set of referred regulations, and the third submenu (Help/Quit) gives choices for help and exit. Figure 18.a illustrates the pull-down menu with three options. Figures 18.a and 18.b illustrate the principal regulation and the referred regulations menu, respectively.

The Prolog commands to access text files in ASCII format are straightforward and no further changes are necessary to be able to access and read the regulatory information. The Prolog command to access the text files is as follows:

```
msg(R,C,S):-
   makestatus(112,"F5 Zoom  Shift-F10 Resize window  Ctrl-F Search  F10 End"),
  makewindow(6,112,120,"",R,C,5,24),
  file_str(S,Datafile),
  display(Datafile),
```

```
Transportation Expert System                    10:37

        Question
 173.403

          r material meet the DOT
          on of a radioactive
          ?
          ecific activity > 70 Bq/gm.
 Check Help for more information
 (Reg 49CFR 173.403)
                              Regulations Related RegsHelp/Quit

        Answer

<CR> to select, Q to exit pulldown menu
```

Fig. 18.a. Pull-down menu with main regulations.

```
Transportation Expert System                          10:37

               173.433   on

Does your      173.423   eet the DOT
definitio                oactive
material?      173.443
(i.e.,spe                ity > 70 Bq/gm.
Check Hel      173.465   information
(Reg 49CF
               173.466          Regulations Related RegsHelp/Quit


                     Answer


<CR> to select, Q to exit pulldown menu
```

Fig. 18.b. Pull-down menu with referenced regulations.

readkey(_),

removewindow,

removestatus.

where **msg(R,C,S)** is a Prolog multicommand that carries information about row (R), column (C), and the name of the file (S). First, the command draws a status windows (to give directions); second, it draws a window where the text is going to be displayed; third, it attaches the string value of the file name (e.g., 171_234.txt) to a symbolic file name; and fourth, it displays the text file. The fifth and sixth commands are for removing both the status and display windows. The file name contains information obtained directly from the downloading mechanism of RegScan. Figure 19 illustrates the retrieval of the text file representing regulation 49 CFR 173.403.

Windows Hypermedia version. The Windows Hypermedia version uses a hypertext mechanism. This is a novel approach that is becoming very popular for new text search mechanisms. This mechanism consists of creating a special area surrounding a text (e.g., a word, a sentence, or a paragraph). It is mouse driven, so by clicking on a "hyperarea" that represents a regulation, for example, the user is presented with the contents of that regulation. These hyperareas not only connect the text with other pieces of text, but they can also connect the user with different types of actions, such as running other programs or bringing other types of information to the screen. Figure 20 illustrates one GUIDE document, including a hypertext button that represents a regulation. By clicking on this button, the text file containing the regulation is brought into the screen. As noted previously, a regulation text file may contain a reference to another regulation. If the user clicks on the text within the hypertext that represents the referred regulation, the corresponding piece of text is brought into the screen in another window for the user to navigate through (Fig. 21).

4.2.2 Implementation of Graphic Files Access

A great deal of information can be obtained by showing a graphic illustration of a specific situation. In this particular case, there are a fairly large number of packages that meet the qualifications for transporting radioactive materials. For example, the "DOE Evaluation Document for DOT 7A Type A Packaging," published in March 1987 covers a

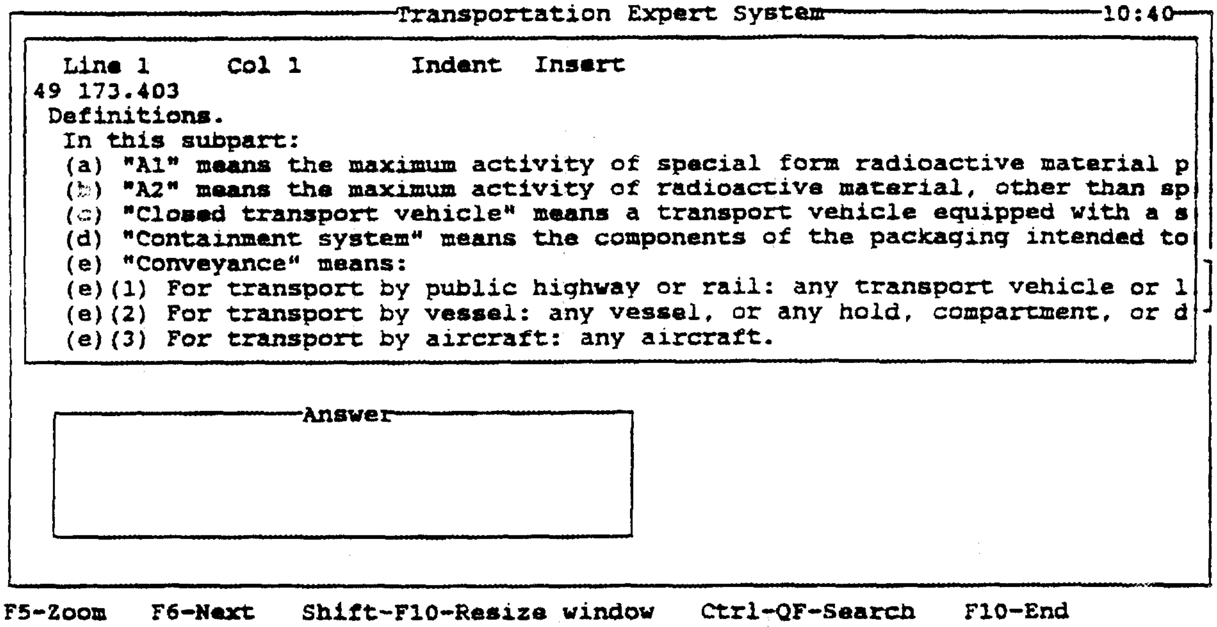

Fig. 19. Prolog retrieval of text files.

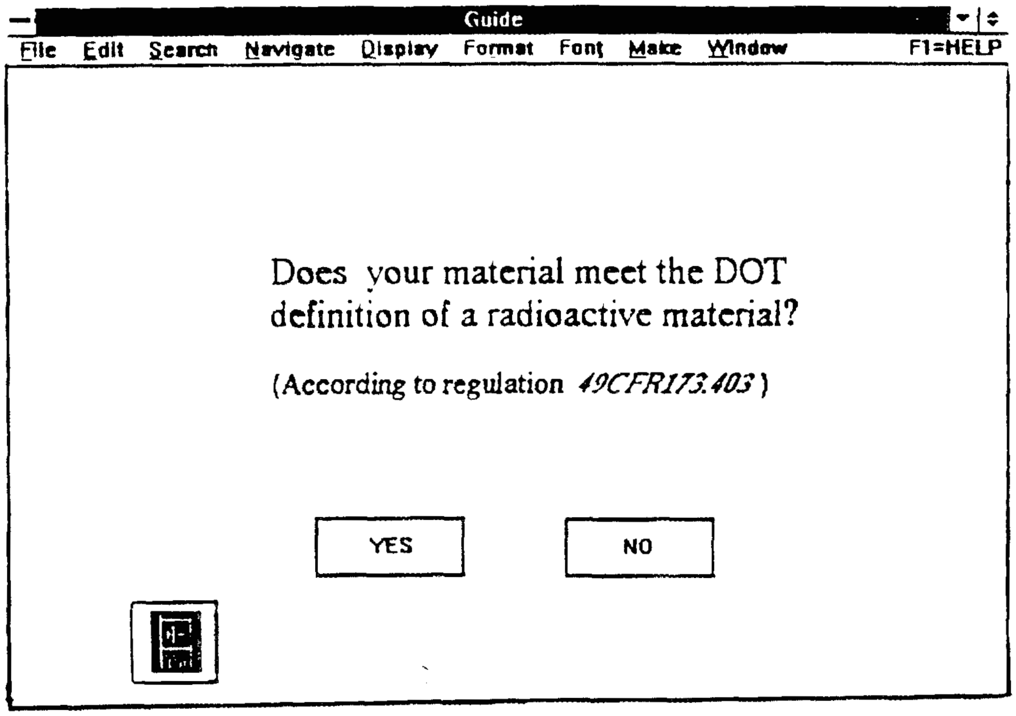

Fig. 20. GUIDE document with a button representing a regulation.

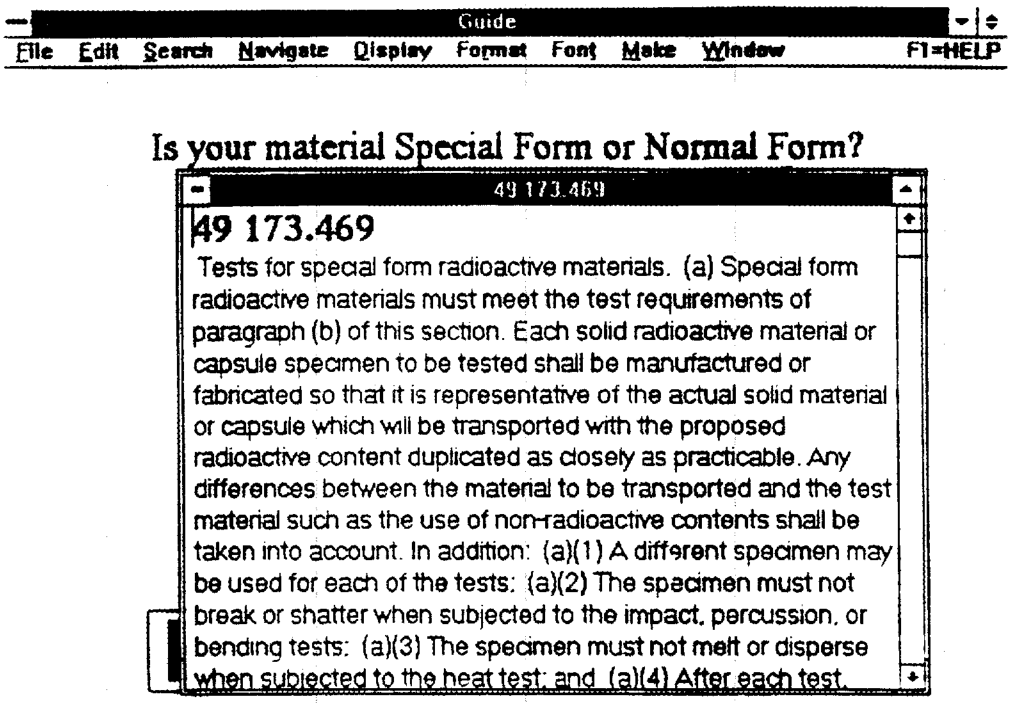

Fig. 21. Retrieval of a text file by a GUIDE document.

wide variety of packages that conform with the definition of a Type A package. Among these types of packages are steel drums, steel boxes, wooden boxes, fiberboard containers, and UF6 cylinders. Every package in this document is illustrated with a figure. Figure 22 shows the illustration of package DOT 7A, Type A.

Electronic copies of the graphic illustrations must be accessible from the expert system prototype. The document mentioned earlier only provides hard copies of the diagrams. Onefairly expeditious way of solving this problem is to scan the image of the document and save the drawing in a format that allows further retrieval from either the DOS or the Windows Hypermedia version. It is possible to obtain a scanned image in bitmap format directly from the image scanner device. This electronic image can be sent to PaintBrush for improvement of the textual information and the addition of colors that highlight details of the drawing.

DOS version. The bitmap file with graphical information has to be accessed by the Prolog program. There is not a direct mechanism to retrieve bitmap files with Prolog commands. However, Prolog can be linked with a C program that does access bitmap files. During the development of this project, a C program was created to read bitmap files in a DOS environment. This C program was then linked to the Prolog program that allowed instructions to be sent from the Prolog environment and retrieve the bitmap files.

Hypermedia version. The different packages presented to the user are displayed first in a menu from where the user can select the one that fits his/her interest. Once the package of choice has been analyzed, the user can always return to the menu to analyze a different package. This process is illustrated in Figs. 23.a and 23.b.

4.2.3 Implementation of the rule-based mechanism

The knowledge of acquisition activities for the development of the prototype resulted in a series of logic diagrams representing decision mechanisms that convey the selection of transportation packages. Every portion of the logic diagram is computerized by using a set of rules written in Prolog. These sets of rules are analogous for both the

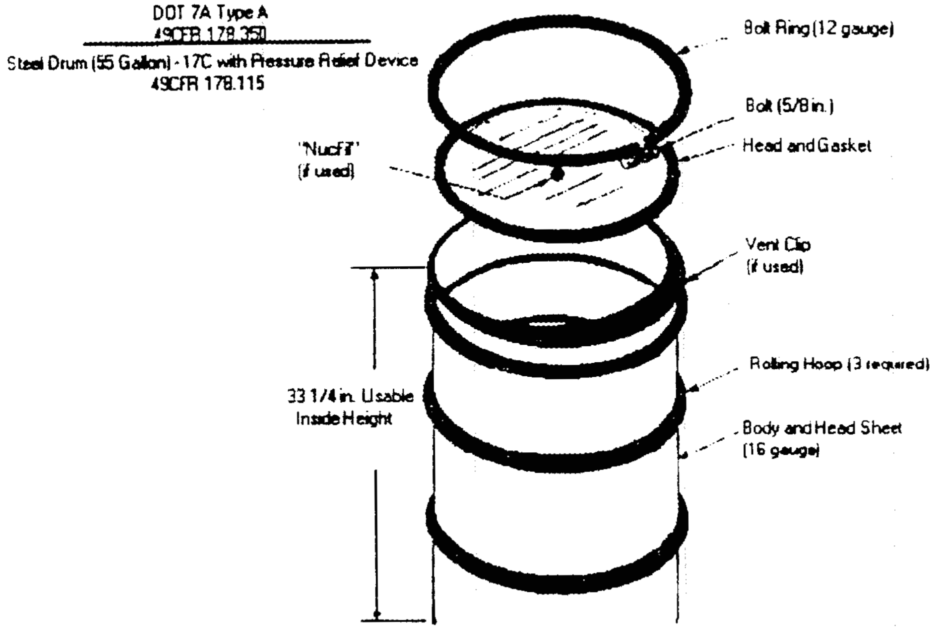

Fig. 22. Graphic file of package DOT 7A, Type A.

```
Transportation Expert System                                10:45

The selected package is Type A package.  Next I'll
show a description of Packages A and graphical description
 of Packages A

Description of Package A
  Line 1     Col 1        Indent  Insert

                TYPE A PACKAGE DESCRIPTION
                **************************

F5-Zoom   F6-Next   Shift-F10-Resize window   Ctrl-QF-Search   F10-End
```

Fig. 23.a. Selection of packages.

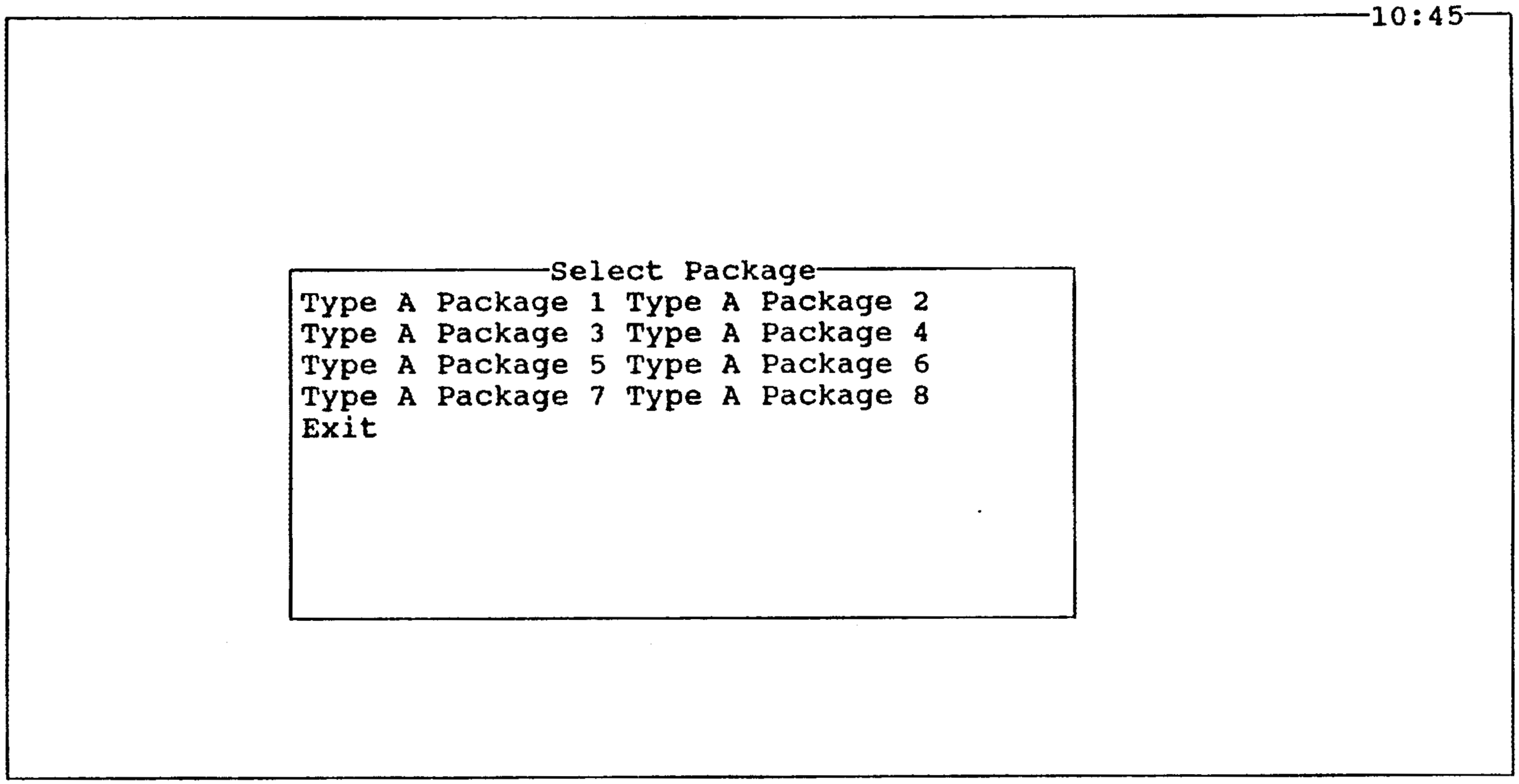

Fig. 23.b. Selection of a package menu.

DOS and Hypermedia versions. A few differences in implementation will be discussed later in this section.

The set of rules that represent the determination of whether a material is fissile will be described as an example of the transformation of a logic diagram into a rule-based system (Fig. 12). The other segments of the logic diagram are similarly transformed.

The first element that the system must evaluate involves determining whether there are potential fissiles in the material to be transported. This is accomplished with the rule **check_for_Fiss(Answer)**

```
check_for_Fiss(IsItFiss):-
    hasfiss(FissTruth),FissTruth="no",
    IsItFiss="NONFS",!.
check_for_Fiss(IsItFiss):-
    expfiss(FissTruth),FissTruth="yes",
    preg_yesto_Expl(IsItFiss),!.
check_for_Fiss(IsItFiss):-
    preg_noto_Expl(IsItFiss),!.
```

This rule has three instances. The first instance of the rule determines whether the material has fissiles with the rule **hasfiss(Answer)** that has four instances of its own.

```
hasfiss(Fisshas):-
    expfiss(FissTruth),FissTruth="no",
    noexpfiss(NonExpTruth),NonExpTruth="no",
    Fisshas="no",!.
hasfiss(Fisshas):-
    expfiss(FissTruth),FissTruth="yes",
    noexpfiss(NonExpTruth),NonExpTruth="yes",
    Fisshas="yes",!.
hasfiss(Fisshas):-
    expfiss(FissTruth),FissTruth="yes",
```

```
    noexpfiss(NonExpTruth),NonExpTruth="no",
    Fisshas="yes",!.

hasfiss(Fisshas):-
    expfiss(FissTruth),FissTruth="no",
    noexpfiss(NonExpTruth),NonExpTruth="yes",
    Fisshas="yes",!.
```

The first instance of **hasfiss(Fisshas)** checks the data bases **expfiss(FissTruth)** and **noexpfiss(NonExpTruth)** that tell if these materials are explicit or nonexplicit potential fissiles. This first instance of the rule returns a nonfissile answer if there are not explicit and nonexplicit fissiles. The second rule for **hasfiss(Fisshas)** returns a yes for fissiles if the materials have explicit and nonexplicit fissiles. The third rule for **hasfiss(Fisshas)** returns a yes for fissiles if the materials have explicit fissiles. The fourth rule returns a yes to fissiles if the materials have nonexplict fissiles.

Once the **hasfiss(Fisshas)** rules have determined the existence or nonexistence of fissiles, the rule instances for **check_for_Fiss(IsItFiss)** can continue. The first instance of this rule is for those cases where there are not fissiles and the program returns a non fissile answer. The second instance of the rule is for those cases in which there are explicit fissiles. The third rule is for those cases in which there are nonexplicit fissiles. The analysis of the second instance of **hasfiss(Fisshas)** indicates that when an explicit fissile exists another rule, **quest_yesto_expl(Answer),** has to be activated before this instance of a rule can return an answer.

```
quest_yesto_Expl(IsItFiss):-
    massexpfiss(MassFiss),MassFiss<=15,
    IsItFiss="FISEXC",!.
quest_yesto_Expl(IsItFiss):-
    questionP1(Answer),
    Answer="yes",
    pregunta_yes_Uenr(IsItFiss),!.
```

```
quest_yesto_Expl(IsItFiss):-
    plutexclus(PlutExclusAns),PlutExclusAns="plutnoexclusive",
    preg_yesto_plutnoexc(IsItFiss),!.

quest_yesto_Expl(IsItFiss):-
    questionP8(Answer),
    Answer="yes",
    IsItFiss="FISEXC",!.
quest_yesto_Expl(IsItFiss):-
    questionP4(Answer),
    Answer="yes",
    IsItFiss="FISEXC",!.
quest_yesto_Expl(IsItFiss):-
    questionP5(Answer),
    Answer="yes",
    IsItFiss="FISEXC",!.
quest_yesto_Expl(IsItFiss):-
    IsItFiss="FISSILE",!.
```

There are seven instances of the rule **quest_yesto_Expl(IsItFiss)** from which an answer can be obtained. The first instance of the rule checks in the data base, **massexpfiss(MassFiss)**, to determine if the mass of fissiles is less than 15 g. If it is less than 15 g, then the rule is fired and it returns the value FISEXC (fissile excepted) that goes into the rule **check_for_Fiss(IsItFiss)**, which returns the same value to the rest of the program. The second instance of **quest_yesto_Expl(IsItFiss)** for those cases when the mass is larger than 15 g, asks **questionP1(Answer)** to the user.

```
questionP1(Answer):-
    shift3_4_retr,nl,
    write(" Does package contain uranium enriched in U-235 to a max."),nl,
    write(" of 1% by mass AND mixed with a total plutonium and U-233"),nl,
    write(" content of up to 1% of mass U-235?"),
    shiftwindow(4),clearwindow,nl,
```

```
write("        (Please, answer yes or no)"),nl,nl,
readln(Answer).
```

A positive answer from the user sends the program to the rule **question_yes_Uenr(IsItFiss).**

```
question_yes_Uenr(Respuesta):-
    questionP2(Answer1),
    Answer1="yes",
    Respuesta="FISEXC",!.
question_yes_Uenr(Respuesta):-
    slg(SLG),SLG="liquid",
    preg_yes_toliq(Respuesta),!.
question_yes_Uenr(Respuesta):-
    questionP4(Answer),
    Answer="yes",
    Respuesta="FISEXC",!.
question_yes_Uenr(Respuesta):-
    questionP5(Answer),
    Answer="yes",
    Respuesta="FISEXC",!.
question_yes_Uenr(Respuesta):-
    Respuesta="FISSILE",!.
```

This rule has five instances. The first instance asks the **questionP2**

```
questionP2(Respuesta):-
    shift3_4_retr,nl,
    write(" Will your fissile radionuclides be distributed homogeneously"),nl,
    write(" throughout the package contents and not form a lattice"),nl,
    write(" arrangement within the package?"),
    shiftwindow(4),clearwindow,nl,
```

```
        write("      (Please, answer yes or no)"),nl,nl,
        readln(Respuesta).
```

A positive answer to this question from the user causes the rule to return a value FISEXC (fissile excepted). A negative answer sends the program to the second instance of **question_yes_Uenr(Respuesta)**, which checks in the data base **slg(SLG)** if the material is a liquid. If the substance is liquid, the program goes to rule **quest_yes_toliq(Answer)**.

```
quest_yes_toliq(Answer):-
    questionP6(Answer1),
    Answer1="yes",
    Answer="FISEXC",!.
quest_yes_toliq(Answer1):-
    questionP4(Answer),
    Answer="yes",
    Answer1="FISEXC",!.
quest_yes_toliq(Answer):-
    questionP5(Answer1),
    Answer1="yes",
    Answer="FISEXC",!.
quest_yes_toliq(Answer):-
    Answer="FISSILE",!.
```

The rule **quest_yes_toliq(Answer1)** has four instances. The first instance asks the **questionP6(Answer)**.

```
questionP6(Answer1):-
    shift3_4_retr,nl,
    write(" Does your material consist of liquid solutions of uranyl nitrate"),nl,
    write(" enriched in U-235 to a maximum of 2% by mass, with total plutonium"),nl,
    write(" and U-233 content not exceeding 0.1% of the mass of U-235 with"),nl,
    write(" a nitrogen-to-uranium atomic ratio (N/U) of 2?"),
    shiftwindow(4),clearwindow,nl,
```

```
        write("        (Please, answer yes or no)"),nl,nl,
        readln(Answer1).
```

A positive answer from the user causes the instance to return a value of FISEXC (fissile excepted). A negative answer takes the program to the second instance of the rule that asks **questionP4(Answer)**, which returns a value of FISEXC (fissile excepted).

```
questionP4(Answer):-
        shift3_4_retr,nl,
        write(" Will your package contain no more than 5 grams of fissile"),nl,
        write(" radionuclides in any 10 liter volume, and will this limit"),nl,
        write(" be maintained throughout transport?"),
        shiftwindow(4),clearwindow,nl,
        write("        (Please, answer yes or no)"),nl,nl,
        readln(Answer).
```

A negative answer takes the system to the third instance of the rule and asks **questionP5(Answer)**. A positive answer from the user will cause the program to return a value of FISEXC (fissile excepted). A negative answer will cause the program to return a value of FISSILE.

If the answer for the second instance of **question_yes_Uenr(Answer)** is not a liquid, the program goes to the third instance of this rule that asks **questionP4(Answer)** (shown above). A positive answer returns a value of FISEXC (fissile excepted). A negative answer returns a value of FISSILE.

This finalizes the second instance of **quest_yesto_Expl(IsItFiss)** for the instance of **question_yes_Uenr(Respuesta)** that was activated with a positive answer for **questionP1(Answer)**. A negative answer to **questionP1(Answer)** sends the program to the third instance of this rule which checks the data base, **plutExcl(PlutExc)**, to determine if plutonium is not exclusive. If it is not exclusive, it goes to rule **preg_yesto_plutnoexc(IsItFiss)**.

```
preg_yesto_plutnoexc(IsItFiss):-
    questionP6(Answer1),
    Answer1="yes",
    IsItFiss="FISEXC",!.
preg_yesto_plutnoexc(IsItFiss):-
    questionP4(Answer),
    Answer="yes",
    IsItFiss="FISEXC",!.
preg_yesto_plutnoexc(IsItFiss):-
    questionP5(Answer),
    Answer="yes",
    IsItFiss="FISEXC".
preg_yesto_plutnoexc(IsItFiss):-
    IsItFiss="FISSILE".
```

The first instance of this rule asks **questionP6(Answer)** as shown above. A positive answer returns a value of FISEXC (fissile excepted). A negative answer sends the program to the second instance of the rule that asks **questionP4(Answer)** as shown above. If the answer is positive, it returns a value of FISEXC (fissile excepted). A negative answer will take the program to the third instance of this rule where it asks **questionP5(Answer)**. A positive answer returns a value of FISEXC. A negative answer sends the program to the fourth instance of this rule that returns a value of FISSILE.

If the system finds that it has a plutonium exclusive material, then the fourth instance of **quest_yesto_Expl(IsItFiss)** is accessed. It asks **questionP8(Answer)**.

```
questionP8(Answer):-
    shift3_4_retr,nl,
    write(" Does 20% or less by mass consist of Pu-230 or Pu-241?"),nl,
    shiftwindow(4),clearwindow,nl,
    write("       (Please, answer yes or no)"),nl,nl,
    readln(Answer).
```

A positive answer returns a value of FISEXC. A negative answer takes the program to the fifth instance of **quest_yesto_Expl(IsItFiss)** that asks **questionP4(Answer)** as shown above. A positive answer returns a value of FISEXC. A negative answer takes the user to the sixth instance of the rule that asks **questionP5(Answer)** (shown above). A positive answer takes the user to the seventh instance of the rule that returns a value of FISSILE.

These sets of rules have answered the second instance of **check_for_fiss(Fiss)** for those cases in which the material has explicit fissiles. If the material does not have explicit fissiles, then the third instance of this rule is accessed. This instance sends the user to **preg_noto_Expl(IsItFiss),** which has five instances.

```
preg_noto_Expl(IsItFiss):-
    questionQ1(Answer),
    Answer="yes",
    IsItFiss="FISEXC",!.
preg_noto_Expl(IsItFiss):-
    nonexplfiss(Enrich),Enrich="less5",
    goforless5(IsItFiss).
preg_noto_Expl(IsItFiss):-
    questionP4(Answer),
    Answer="yes",
    IsItFiss="FISEXC",!.
preg_noto_Expl(IsItFiss):-
    questionP5(Answer),
    Answer="yes",
    IsItFiss="FISEXC".
preg_noto_Expl(IsItFiss):-
    IsItFiss="FISSILE)",!.

questionQ1(Answer):-
    shift3_4_retr,nl,
    write(" Are you carrying 15 grams or"),nl,
    write(" less of fissile radionuclide?"),
```

```
        shiftwindow(4),clearwindow,nl,nl,
        write(" (Please, answer yes or no)"),nl,
        readln(Answer).
```

The first instance asks the questionQ1(Answer). A positive answer returns a value of FISEXC. A negative answer sends the program to the second instance of the rule. This instance checks the data base, **nonexplfiss(NONEXP)**, if the material is enriched less than 5%. If this search is true, the program goes on to **goforless5(IsItFiss)**, which has five instances.

```
goforless5(IsItFiss):-
    questionP7(Answer),
    Answer="yes",
    goforUenr1(IsItFiss).
goforless5(IsItFiss):-
    slg(SLG),SLG="liquid",
    goforless5liq(IsItFiss).
goforless5(IsItFiss):-
    questionP4(Answer),
    Answer="yes",
    IsItFiss="FISEXC",!.
goforless5(IsItFiss):-
    questionP5(Answer),
    Answer="yes",
    IsItFiss="FISEXC".
goforless5(IsItFiss):-
    IsItFiss="FISSILE".
```

These instances and the derived rule instance are handled in the same manner as the other rules. The same mechanism of backtracking until it finds an acceptable answer is followed throughout the analysis of this and other rule-based systems that represent the logic diagram.

DOS version. The prototype has been developed in Prolog. Several modules comprise the entire program. A main program that has the ability to gather information such as isotopes, activity, physical state, etc., is linked to other rule-based modules and a graphic retrieval program. Figure 24 illustrates the different segments of the main program and the respective rule-based systems that are linked to it. The first rule-based program linked to the main program is the fissile determination module. The second linked program is the limited quantity determination module. The third program is the LSA/SCO determination module. The fourth program is not a rule-based program, but a straightforward C program that can read bitmap graphic files and is activated from within the main program with the following Prolog command:

```
show(Package):-
    initgraph(detect,2,_,_,""),
    setviewport(1,1,438,479,1),
    setbkcolor(blue),
    statusLine("Please, wait, when finished press a key to continue"),
    display_bitmap(Package),
    readchar(_),closegraph.
```

This Prolog command initializes the graphic environment, sets the view port, the color of the background, and then, using the command **display_bitmap(Package)**, the bitmap file is accessed. The command **display_bitmap(Variable)** is a C function that has been linked to Prolog.

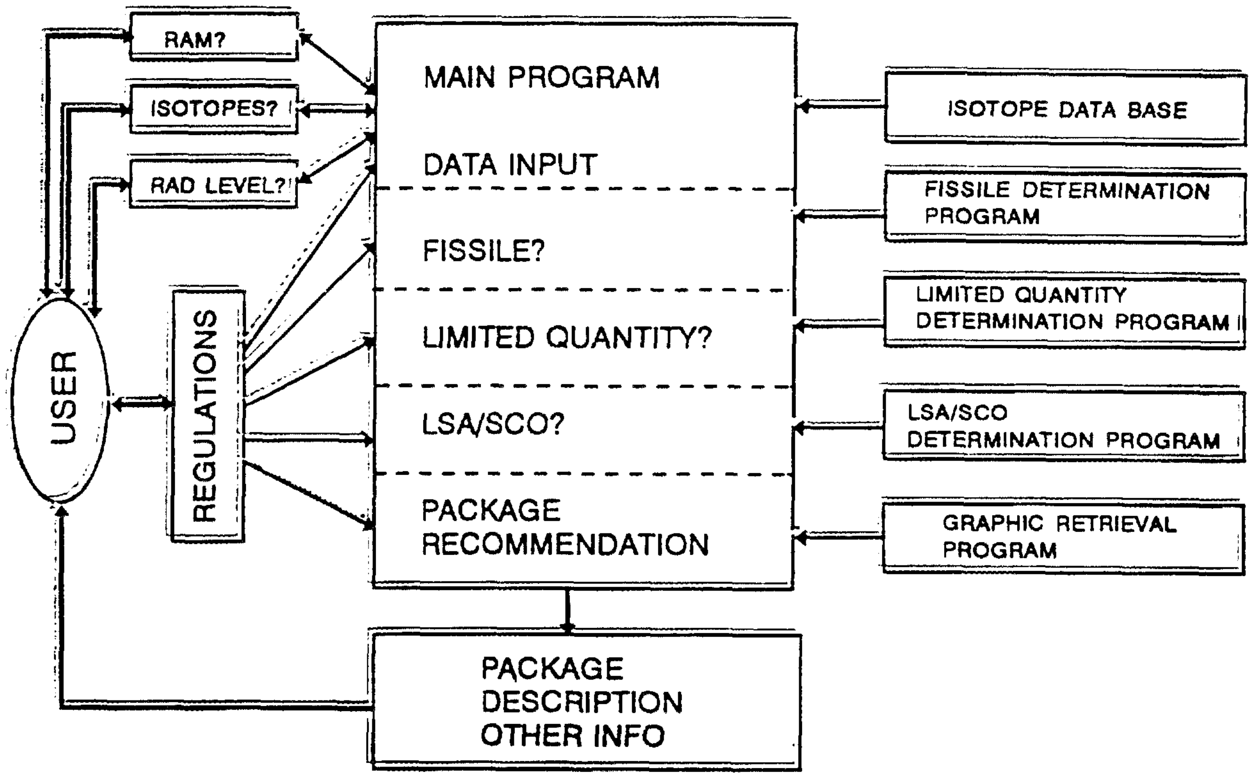

Fig. 24. Main program architecture for the DOS version.

The main program also accesses data bases of isotopes and their characteristics and text files of (1) the transportation regulations, and (2) the package descriptions.

Hypermedia version. The GUIDE document works within a Windows 3.0 environment. Initial questions start a consultation session. Regulations can be checked and navigated using the hypertext mechanism. If required, the GUIDE document can access a rule-based module to obtain information about isotopes or activities by launching the execution of the input data module. Once the program is launched, a user interaction program starts working under the DOS environment (even though GUIDE works under Windows). The other rule-based modules accessed by the GUIDE document are the fissile determination, the limited quantity determination, and the LSA/SCO determination programs. These modules are identical to the DOS version.

4.2.4 Implementation of a Friendly Interface

There is a need for a close interaction between the program and the user. The system has to be able to present the user with questions in a clean and direct manner. The system is also required to provide a simple mechanism for the user to answer the questions. In addition, access to additional information such as text files, data bases, or graphics has to be invisible to the user.

DOS version. The interactive mechanism provided by Prolog is mainly comprised of text windows and menus. Figure 25 illustrates the mechanism by which the system asks the user if the material meets the DOT definition of a radioactive material. It displays the question in a text window format. Before the user can answer the question, the system provides the user with the regulations. By pressing the return key in Regulations, for example, the program pops a menu from which to select a regulation (Fig. 26).

In order to answer the question about isotopes and activity of each element, the system uses a multiple choice scroll-down menu that presents the user with the isotopes existing in the data base. The user can select more than one isotope. After selecting isotope(s), the program asks for the activity of the selected isotope(s). Figure 27 shows

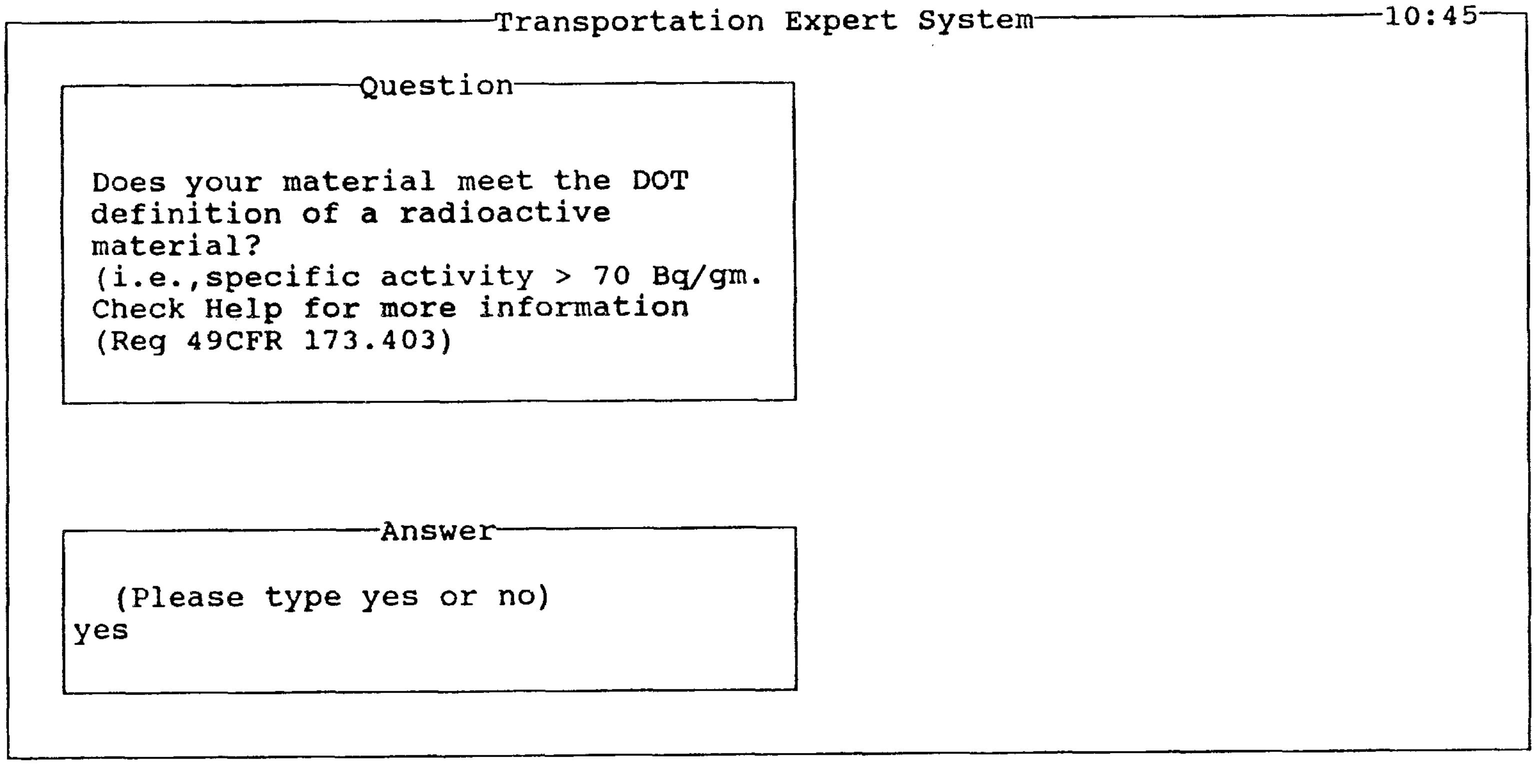

Fig. 25. Interaction between the user and the program.

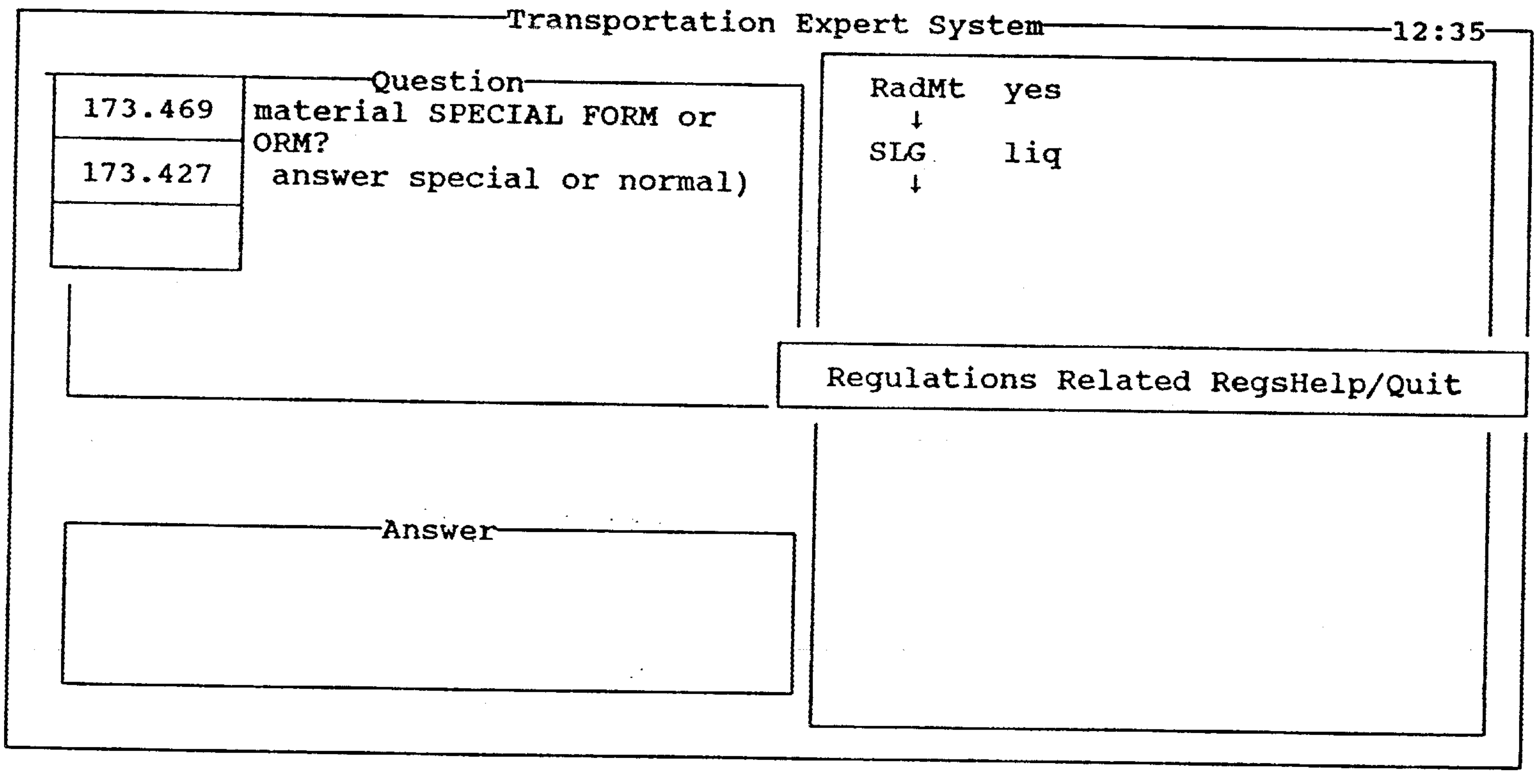

Fig. 26. Regulation selection menu.

```
Transportation Expert System                                 10:43

  What isotope do you want to ship          Select
                                            europium-152
  (Select from the following list)          europium-154
                                            europimu-155
                                            radium-223
                                            radium-224
                                            radium-226
                                            uranium-232
                                            uranium-233
                                            uranium-235
                                            uranium-236
                                            cobalt-58
                                            cobalt-57
                                            californium-252
                 Answer                     iron-55
                                            h-3
                                            iridium-192
                                            iodine-123

Please select with the return key, F10 when selection is made
```

Fig. 27. Multiple choice menu.

the multiple choice menu, while Fig. 28 illustrates the questions concerning the activity content. The DOS version also records the path of the decision stages (i.e., whether the material isradioactive, solid, and a special form material). The system calculates that if the fraction of A_2 is above 1, determines that it is nonfissile, and that it has not qualified as limited quantity. A menu from which to select packages is also used (Fig. 29).

Hypermedia version. The interactive mechanisms offered by the GUIDE documents consist of hypertext buttons that may be attached to text files, commands, or graphic files. Each button accesses different features according to what kind of command is attached. Figure 30.a illustrates a GUIDE document referring to a specific regulation, Fig. 30.b illustrates a GUIDE document accessing the text file representing a specific regulation, and Fig. 30.c illustrates the navigation procedure through several referenced regulations.

In Fig. 31.a, a GUIDE document is illustrated. By clicking on any button in this document, the user is able to access a graphics file. Figure 31.b illustrates the display of a graphic file with information about a specific Type A package. Most of the features pertaining to GUIDE are normally driven by the use of buttons. All of these buttons are linked to text files, which can, in turn, be linked to other text, graphic files, or to LOGiix commands that order the system to perform a certain action. The next portion of a LOGiix command refers to the execution of the first rule-based program:

```
#LOGiix
function main()
Begin
    Launch("forguid1.exe","",1);
    result := answer(0+64,"Continue","Press OK to Continue");
    if (result = 1 ) then
    NextFrame(ButtonDoc());
End
```

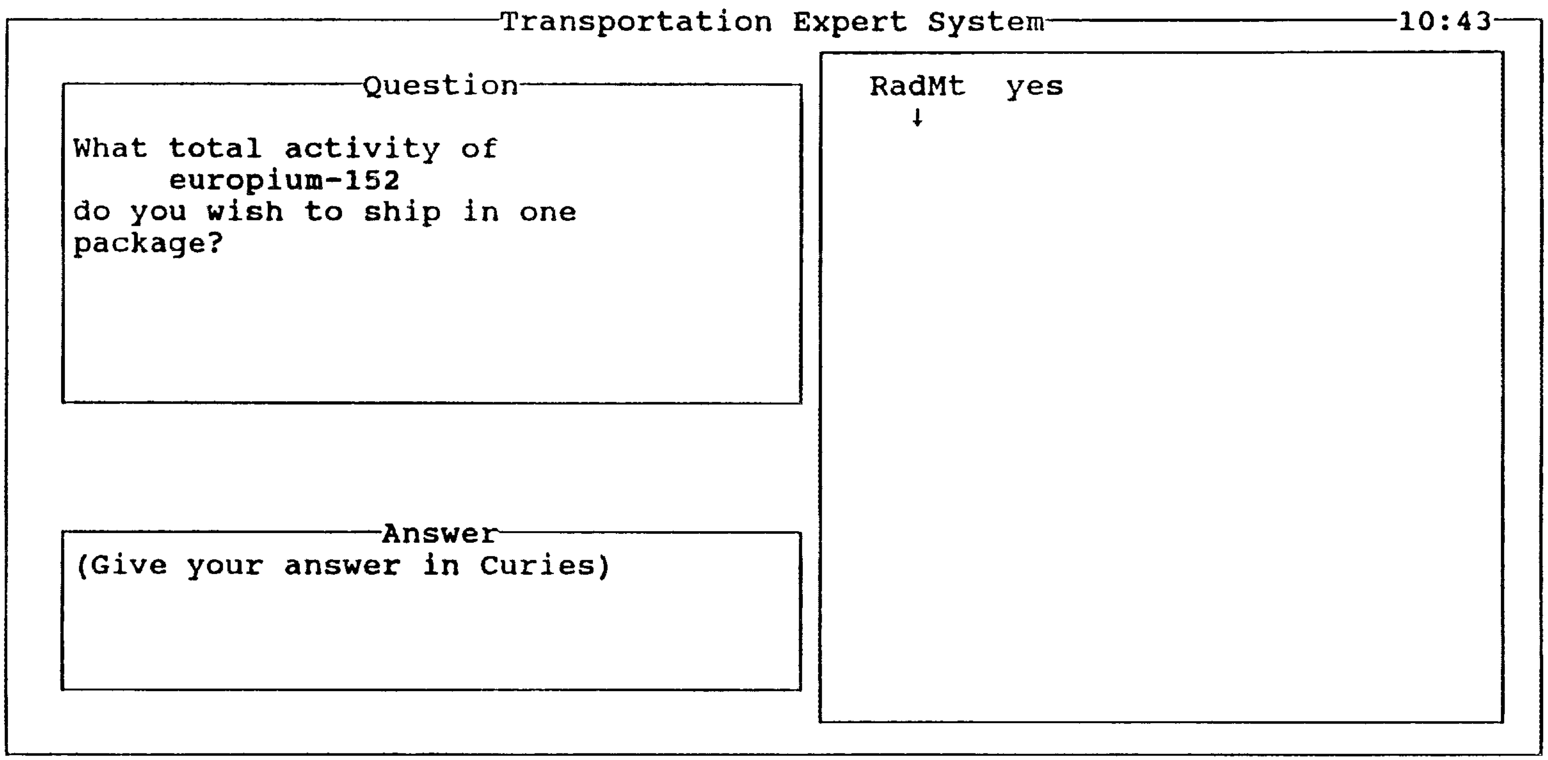

Fig. 28. An example of a question-answer interactive window.

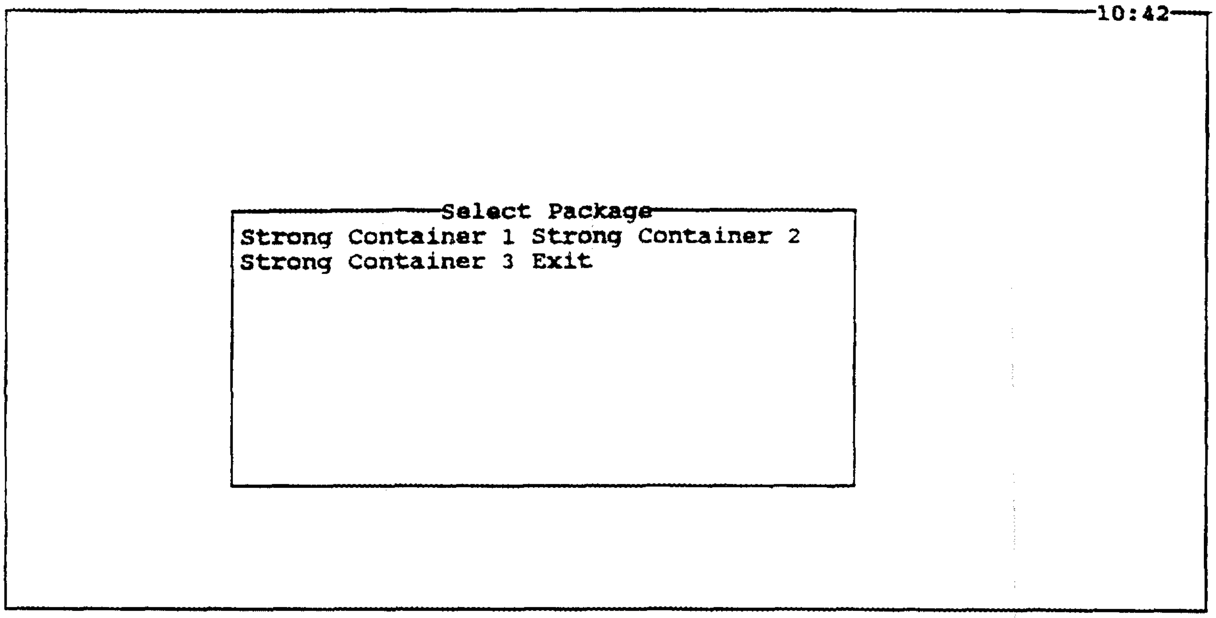

Fig. 29. Menu for the selection of packages.

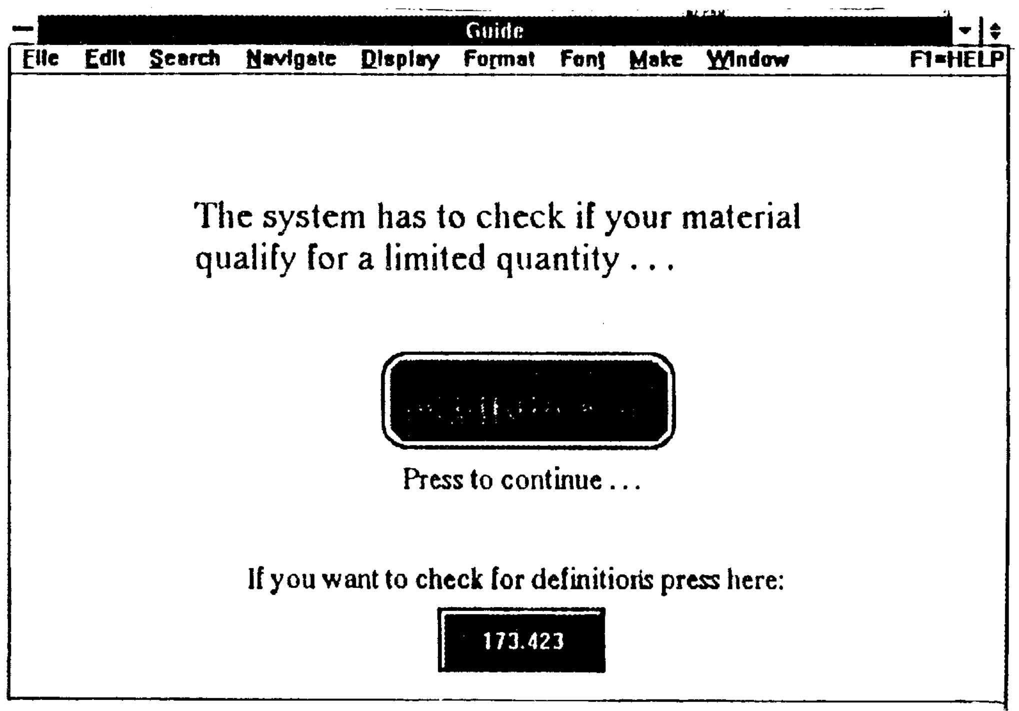

Fig. 30.a. GUIDE pointing to a regulation text file.

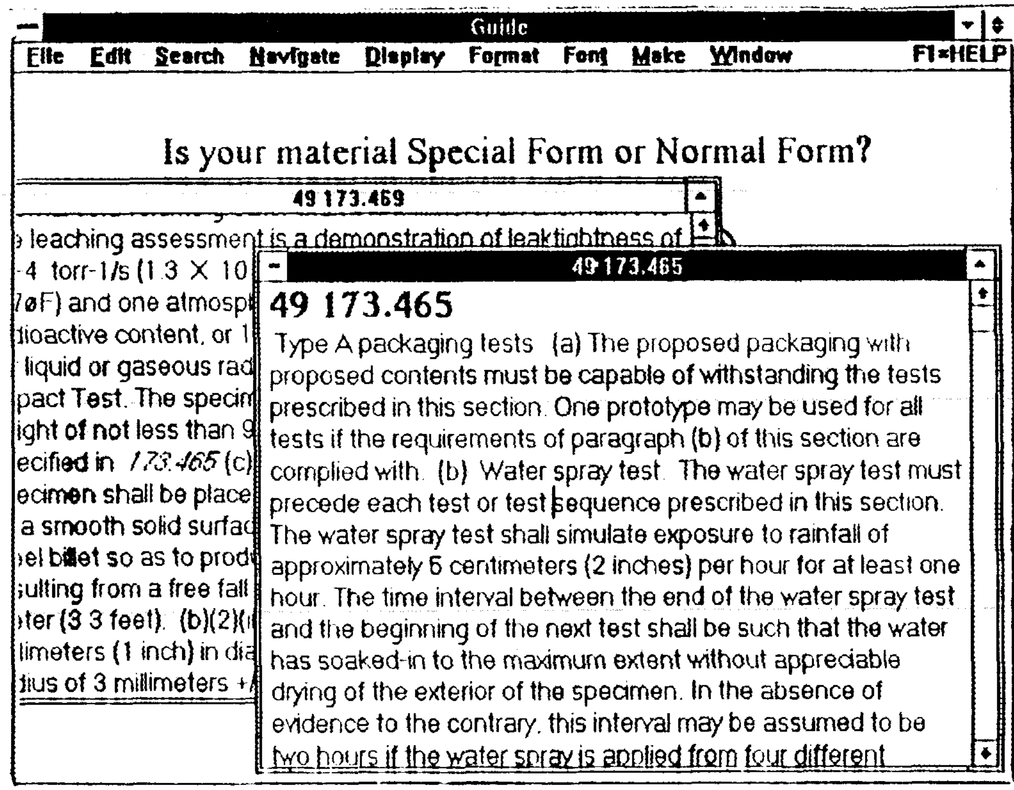

Fig. 30.b. Access to text files.

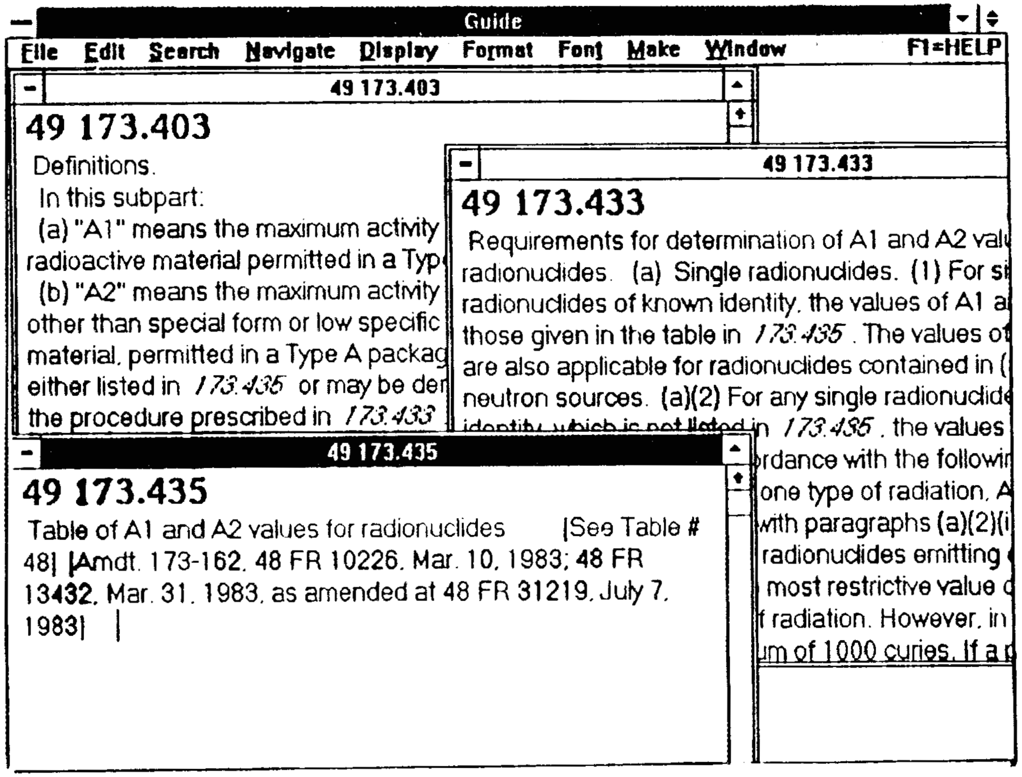

Fig. 30.c. Navigation through text files.

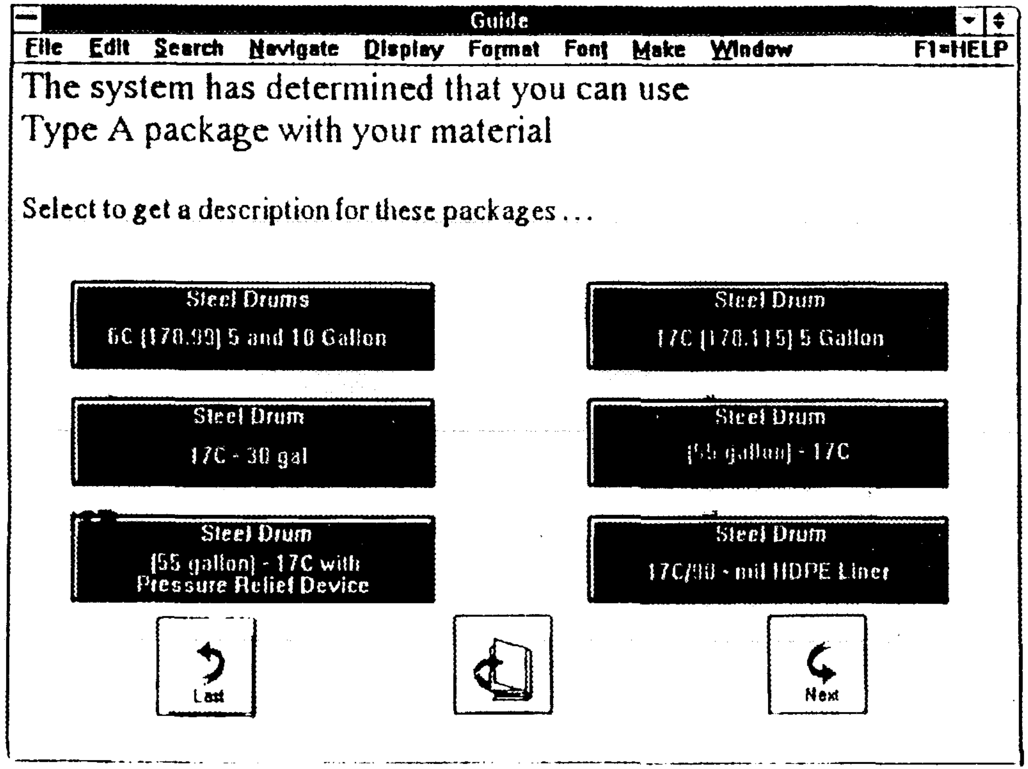

Fig. 31.a. Package selection buttons.

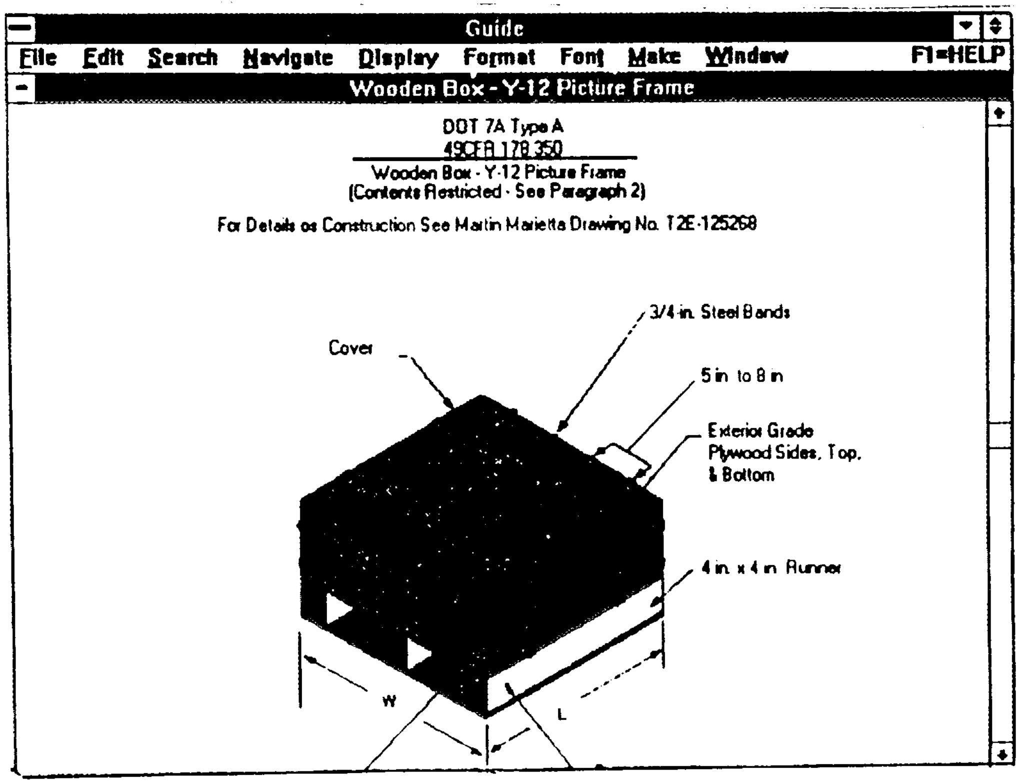

Fig. 31.b. Package graphic display.

By pressing the corresponding hyperbutton, this LOGiix function is initiated. First, it executes the rule-based program, **forguid1.exe**, and by clicking the Continue button, the program goes to the next frame.

The next command is also attached to a button, and depending on the result of the selected action, the command will activate other GUIDE documents:

```
#LOGiix
Function main()
Begin
    Launch("forguid4.exe","",1);
    result:=answer(0+64,"Continue:,"Press OK to continue");
    if (result = 1) then
begin
    type := ReadFromFile ("file7.txt");
    messagebox("type =", type;
    close(ButtonDoc(),1+256);
    if (type = "typea") then
    NameOpen("typeaop.gui")
else
    if (type = "typeb") then
    NameOpen("typebop.gui");
end
```

This function will execute the rule-based **forguid4.exe**. This program will produce a value that will be written in **file7.txt**. Next, GUIDE will read this value from **file7.txt**, (if the value is **typea**), then GUIDE will open another GUIDE document, **typeaop.gui**. If the value is **typeb**, GUIDE will open the GUIDE document **typebop.gui**. Both of these files will show graphics and video files of the respective package type.

4.2.5 Access to Audiovisual Information

Modern days have brought to life extraordinary sources of information. Descriptions of objects, as well as actions that need to be performed on those objects, normally required long pieces of text to convey the message. Fortunately, video media have been developed for computers. A great deal of information can be put in a piece of recording element (such as a video tape). All of this information can be stored in a short period of time. However, the video media are not a substitute for the written material that accompanies a description, but they are a formidable complement. With both written and video documents, a description can be very clear and straightforward. Difficult or lengthy explanations can be abbreviated with well-designed sequences of video images. The text file can be used to describe the elements that the user has to pay special attention to, and the video information can describe everything else in a video-motion set of images.

DOS version. Display of video images with the Prolog DOS version is still in the developmental stage. The video-motion card can be acquired with essentially two types of software. One type of software allows Windows 3.0 interaction, and the other type has utilities that allow access to video images within the DOS environment. The mechanism is simple and consists of writing a series of commands to a text file. As soon as these commands are written to the text file, the utilities of the video intercept them and perform the commands on the video equipment. The image from the video equipment is neat and clear. However, resizing of the video image has not yet been possible. In addition, the commands to control the video equipment have not been established. This means that the allocation of a sector within a video tape is not accessible, nor are the features of a video player (such as play, stop, or rewind).

The commands for playing a portion of a video image sequence within a window can be simply obtained using the following Prolog commands:

```
show_video:-
        makewindow(1,87,52,"Transportation Expert System",5,5,13,48),
        openwrite(myfile1,"mic"),
    writedevice(myfile1),
```

```
write("system on;"),
write("tran 5;"),
write("video on;"),
write("audio on;"),
flush(myfile1),
readchar(_),
    write("system off;"),
flush(myfile1),
closefile(myfile1).
```

The first line of this Prolog rule opens a window where the video image sequence is going to be displayed. The second line opens a file **mic** in which a few commands are going to be written. The third line prepares the file **mic**. The fourth line writes in the string **system on** in the file. The fifth line writes the string, **tran** 5, and the sixth line writes the command **video on**. Finally, the seventh line writes **audio on**. These are the basic commands to turn the system on, make the window transparent to color 5 (magenta), and display the video image. Subsequently, the command turns the video on first and then the audio. The next step in sending commands to the video equipment is in the developmental stage.

Hypermedia version. A GUIDE document has hyper buttons that send commands to the video equipment and to the video card of the computer. The following LOGiix commands illustrates this procedure:

```
#LOGiix
function main()
begin
    Launch("c:\\micutils\\vcrwin1.exe","c:\\micutils\\tape1.dat 5",1);
end
```

These commands execute the program, **vcrwin1.exe**, which take the second string, **tape1.dat5**, as running parameters. The first parameter is a text file in which each line represents one piece of videotape. Each line contains the number of the piece, length of playing time, length for rewinding, and length for forwarding. The number 5 represents the number of the piece that will play. The program, **vcrwin1.exe**, opens the video window to display the video image sequences and sends commands to control the video equipment.

During the development of the video image access, it was necessary to develop a piece of hardware to control the video equipment, which in this case was a video cassette recorder (VCR). Consequently, it was necessary to create the file, **tape1.dat**, to store the video tape information. Once a laser video disc player is acquired, the operation will be a little different. There will be no need to control the equipment with special hardware, and the access time to each piece of video image sequence will be much faster. In any event, the use of a VCR provides a flavor of what video images can add to an expert system in terms of information and friendliness.

5. ANALYSIS

The feasibility of developing a full expert system for the transportation and packaging of hazardous and radioactive materials has been initiated within the framework of three subtasks: (1) analysis of commercial packages related to regulation scanning, (2) analysis of computer languages to develop the expert system, and (3) development of expert system prototypes. The strategy to develop the latter subtask was to a) develop modules to capture the knowledge of different areas of transportation and packaging, and b) to analyze the feasibility of appending these different modules in one final full package. The individual modules contemplate one prototype for transporting and packaging of radioactive materials, and another for transporting hazardous chemical materials. In the event that it is not feasible to link these two software modules, the modules can always be used as stand-alone tools, or linked as a single package with some restrictions in their applicability. The work done during fiscal year 1992 has focused on developing a prototype for transporting radioactive materials.

The final version of the expert system will be implemented in a computer system to allow a better and less expensive distribution to the final users. The first prototype module has been initialized and will capture the expertise for the transportation and packaging of radioactive materials. The general strategy to develop this module has been to develop a logic diagram of the decision making process for transporting and packaging of radioactive materials, and to put this expertise in Prolog code. The logic diagram has gone through an extensive revision process, and the user interface has almost been defined in its entirety.

The analysis of the commercially available software, RegScan and Environmental/Safety Library, indicated that both packages, although very useful for navigating the pertinent regulations, are not particularly suitable for the determination of the type of packaging required for hazardous and radioactive material transportation. However, the regulations can be downloaded from these software systems and used as the source data base for the regulations that will be accessed by the expert system.

The three computer languages analyzed before show great potential for the development of the expert system. From the analysis of the implementation of the different features required for the expert system prototype, it was concluded that the developmental efforts should be directed to two versions of the prototype. First, a simpler DOS version that could be run on any computer machine to accelerate the review process was recommended. Second, a more sophisticated Windows Hypermedia version that includes every required feature was built as a proof of concept that the entire package can be built. The DOS version of the prototype should be developed using PDC Prolog because the language allows more flexibility with the different features incorporated in it. The distribution of the prototype for revision purposes may be simpler than with the other programs.

The hypermedia version was developed using a combination of GUIDE and Prolog. Hypermedia technology usually works as an interactive software system that gives personal computer users the ability to organize, manage, and present information in a number of formats—text, graphics, sounds, and full-motion video. GUIDE may use LOGiiX, an embedded programming language which allows the user to perform complex information manipulations. One of the manipulations is the possibility of adding the reasoning mechanism developed in the DOS version. GUIDE does not have a rule-based system incorporated in the system which allows the development of expert systems. However, LOGiiX provides commands that can access rule reasoning mechanisms at any point during the consultation.

6. FUTURE WORK

The first phase in demonstrating the feasibility of developing an expert system prototype has produced very promising results. It is possible to represent knowledge about radioactive material transportation packaging using a logic diagram that has been translated into a rule-based system. The features required to assist a user during a consultation session have been implemented in both DOS and Windows 3.0 environments. The mechanism to download updated regulatory information from RegScan has proven to be technically feasible and within the purview of the license agreement. The activities related to the assessment of the required review, validation, and verification process for the operational use of the radioactive material transportation prototype is in the planning stage. Links with the Automated Transportation Management System (ATMS) will begin. One of the subtasks proposed for FY 1993 is the investigation of possible ties of the transportation packaging expert system with ATMS. The knowledge gained from this study will indicate the type of tools and mechanisms necessary to link the expert system with the ATMS.

The second phase of the demonstration will include the feasibility analysis of the hazardous chemical materials transportation packaging expert system prototype. The possibility of developing an expert system for the transportation packaging of hazardous chemical materials to be linked to the radioactive material transportation packaging expert system will be evaluated. The analysis will determine the possibilities of linking the program to the ATMS.

7. REFERENCES

1. Ferrada, J. J., Holmes, J. M., *Chemical Engineering Progress*, (April 1990).
2. Tarnuzzer, A., personal communication, October 1991.
3. Smalltalk/VWindows, *Tutorial and Programming Handbook,* Digitalk, 1991.

Internal Distribution

1. J. M. Begovich
2. C. G. Cowart
3. A. G. Croff
4. J. J. Ferrada
5. R. K. Genung
6. V. M. Green
7. R. D. Halverson
8. K. W. Hoff
9. D. S. Joy
10. K. H. King-Jones
11. R. C. Mason
12. I. W. Osborne-Lee
13. B. D. Patton
14. R. B. Pope
15. R. R. Rawl
16. B. R. Rodgers
17. M. J. Welch
18. Central Research Library
19. ORNL Y-12 Technical Library Document Reference Section

20-21. Laboratory Records (2 copies)

22. Laboratory Records, R.C.
23. ORNL Patent Section

External Distribution

24. L. G. Blalock, Manager, Transportation Logistics Program, Transportation Management Division, U.S. Department of Energy, Washington, D.C. 20585
25. G. Brown, Radioactive Materials Branch, Office of Hazardous Materials Transportation, U.S. Department of Energy, Washington, D. C. 20590
26. W. Carriker, Radioactive Materials Branch, Office of Hazardous Materials Transportation, U.S. Department of Energy, Washington, D. C. 20590
27. G. F. Honstreiter, Sandia Laboratory, Transportation System Technical Division, Albuquerque, NM 87185.
28. R. F. Jarrell, Traffic Manager, Department of Operations, Richland Operations Office, Richland, WA 99352
29. P. B. Lester, Oak Ridge Field Office, U.S. Department of Energy, Oak Ridge, TN 37831-8620.
30. R. E. Luna, Sandia Laboratory, TRA SAD Division, Albuquerque, NM 87123.
31. C. MacDonald, CRWMS M&O, Vienna, VA 22180
32. J. O'Steen, Office of Hazardous Materials Transportation, Department of Transportation, Washington, D.C. 20590.
33. J. Portsmouth, Westinghouse Hanford Company, Richland, WA 99352.
39. F. Punch, Division of Quality Verification, U. S. Department of Energy, Washington, D.C. 20545.
40. H. Selling, Radiation Protection Section, NENS, International Atomic Energy Agency, Vienna, Austria.
41. T. M. Thomas, Transportation Logistics Program, Transportation Management Division, U.S. Department of Energy, Washington, D.C. 20585
42. M. E. Wangler, Acting Director Transportation and Packaging Safety Division, Office of Safety and Quality Assurance, Washington, D.C.
43. Office of Assistant Manager, Energy Research and Development, DOE-OR, P.O. Box 2001, Oak Ridge, TN 37831

44-53. Office of Scientific and Technical Information, P. O. Box 62, Oak Ridge, TN 37831