APPENDIX I-1

SOFTWARE REQUIREMENTS AND DESIGN SPECIFICATIONS FOR THE DOCUMENT DATABASE

APPENDIX I-1

SOFTWARE REQUIREMENTS AND DESIGN SPECIFICATIONS
FOR THE DOCUMENT DATABASE

PURPOSE

This document provides the software requirements and design specifications for the Document Database as described in Task 3 of the *Radiological Assessments Corporation* Technical Proposal for Source Term Calculation and Ingestion Pathway Data Retrieval for the Savannah River Site (SRS) Environmental Dose Reconstruction Project, May 11, 1995 (RAC 1995). This document is intended for review by personnel at the Centers for Disease Control and Prevention (CDC) and software developers under subcontract with the *Radiological Assessment Corporation* (RAC).

GENERAL REQUIREMENTS

The Document Database was developed for the SRS Environmental Dose Reconstruction Project to track documents, which are defined as any material that provides information or data used in the project. The database tracks

- Published internal documents and reports,
- Journal articles and books,
- Unpublished reports, memos, and letters,
- Field or office notebooks and log entries,
- Laboratory analysis reports and associated quality assurance documentation,
- Databases, electronic or otherwise, containing information used in the study (e.g. monitoring data),
- Documented information and data obtained from interviews or phone conversations, and
- other documented information.

QUALITY ASSURANCE PROGRAM REQUIREMENTS

Software design, development, testing, and implementation is subject to the quality assurance (QA) requirements stated in Task 6 of the RAC Technical Proposal. These include the following:

1. A software requirements specification document,
2. A software design document,
3. A software test plan (verification and validation test plan),
4. Verification and validation test results and,
5. Software user's manual.

This document provides the software requirements and design specifications. All database requirements as specified in the contract will be me via a program of review and comparison carried out by at least two *RAC* reviewers. In addition, at least two *RAC* reviewers, and the CDC and external reviewers, will examine the database to ensure that: 1) test data, audio files and other nonessential elements are removed from the database before delivery to CDC; 2) all citations in

RAC reports are correctly listed in the database; 3) cross references between *RAC* reports and source documents are correct; and 4) source document authors are listed correctly.

HOST SYSTEM REQUIREMENTS

The document database software will use the Foxpro$^©$ relational database language and will require a microcomputer operating under MS DOS$^©$, Version 6.0 or higher, Windows$^©$ Version 3.1 or higher, and at least 8MB of extended random access memory, although 16MB is preferable. About 200MB of free hard disk space will be required to load and operate the software.

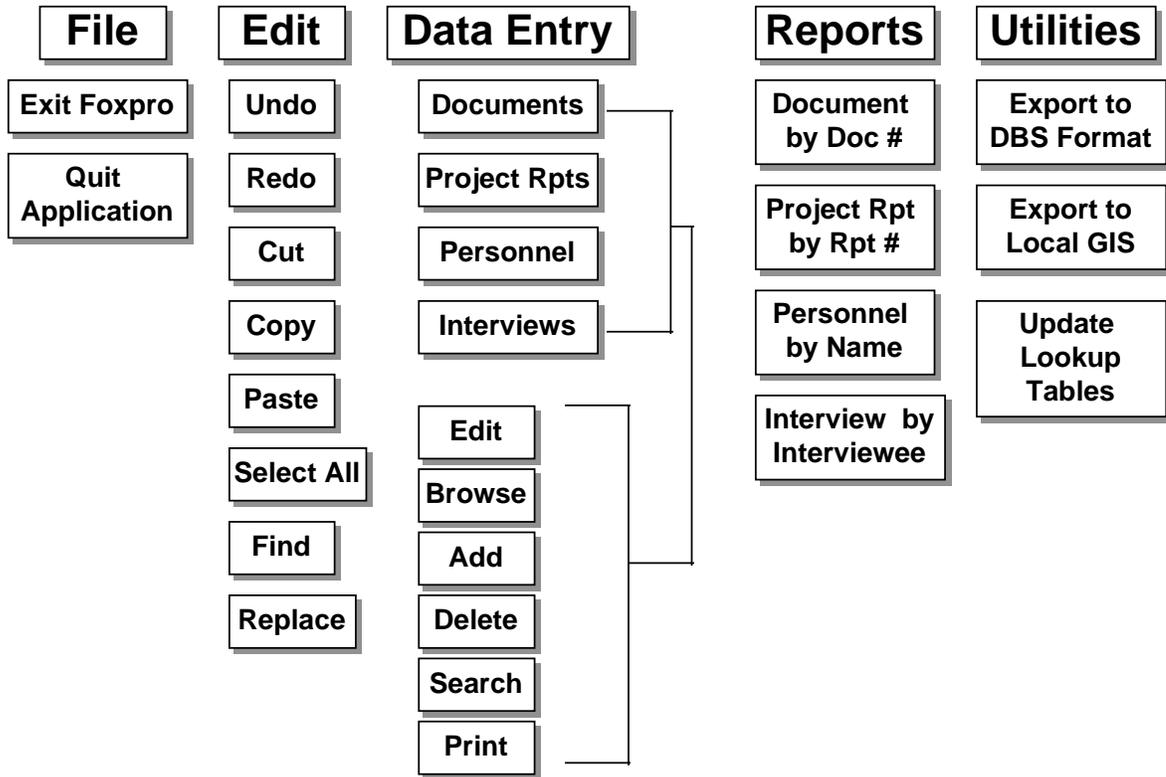## GENERAL USER INTERFACE REQUIREMENTS

The software will provide a user interface to allow a trained nondeveloper to add, subtract, edit, or query fields in the database. The user interface will be written in the Visual Foxpro programming language. The user will be presented with a typical Windows based interface menu. Under each main menu item, the user will be able to select options such as data entry and report generation. The current menu options are shown in Figure I1-1.

Typically, opening, closing, saving, and exiting a Windows application is accomplished through the file menu option. However, this software does not include open, close, and save menu options because the database files will automatically be opened or saved when opening or exiting the application. The *Edit* menu contains the typical Windows options such as *Cut*, *Copy*, *Paste*, and *Select*.

The core of the application is the *Data Entry* menu (Figure I1-1). The user may search for records and view their contents, add records, delete records, and print selected records. Records can be retrieved, added, deleted, edited, or printed. There are four areas of data entry:
1. Those that refer to SRS Environmental Dose Reconstruction discovery documents (*Documents*),
2. Those that contain information of persons involved in the dose reconstruction (*Personnel*),
3. Those that describe documents created for Phase II of the SRS Environmental Dose Reconstruction Project (*Project Reports*), and
4. Those that describe interviews that have been conducted (*Interviews*).

Printed reports can be obtained from the *Reports* menu. Reports may be printed for all the categories listed under the *Data Entry* menu. The *Utilities* menu offers options to export data to other databases via the DBS format, or export the data for use in a Geographical Information System (GIS). The *Utilities* menu also allows access to tables so the lookup table can be updated throughout the life of the project.

File	Edit	Data Entry	Reports	Utilities
Exit Foxpro	**Undo**	**Documents**	**Document by Doc #**	**Export to DBS Format**
Quit Application	**Redo**	**Project Rpts**	**Project Rpt by Rpt #**	**Export to Local GIS**
	Cut	**Personnel**	**Personnel by Name**	**Update Lookup Tables**
	Copy	**Interviews**	**Interview by Interviewee**	
	Paste			
	Select All	**Edit**		
	Find	**Browse**		
	Replace	**Add**		
		Delete		
		Search		
		Print		

Figure I1-1. User interface menus for the Document Database. The *Documents* submenu under the *Data Entry* menu refers to documents found during document searches. *Project Reports* refers to reports generated during Phase II of the SRS Dose Reconstruction Project. The GIS designation under the *Utilities* menu refers to a Geographical Information System.

DESIGN SPECIFICATIONS

The software will use the Visual Foxpro relational database software and programming language. Foxpro databases (designated with the .DBC extension) are composed of tables that are designated with a .DBF file extension. Tables will be created for each *data entity*. Each entity may contain numerous data *fields*. A field contains the information that is either entered or retrieved by the user, or defined based on a logical expression. A field is designated as a character field, numeric field, date field, logical field, or memo field. Character, date, memo, and numeric fields have lengths defined by the developer. That is, the field has a limited number of characters that may be entered in the field by the end-user. Logical fields have a predefined length of 1. Tables are usually linked to other tables by fields of the same name and data type and length. In this way, the information presented to the end-user may originate from several tables that are related by fields of the same type and length.

The user interface will access the tables to enter data, edit existing data, query existing data, and generate reports. The program will be compiled into an executable file for distribution to end-users.

Entity Relationship Specifications

Data entities defined in the Data Dictionary (see Table 1 at the end of this document) are related to one another by the Foxpro generated data entity relationship diagram as illustrated in Figure I1-2 and I1-3.

Referential Integrity

The referential integrity of the database defines how changes made to one table are carried through to all other related tables. Tables are usually defined as parent or child. The parent table is usually where records are added, deleted, or edited. Changes made to the parent table are passed to the child table according to the rules defined in the Foxpro referential integrity builder (Figures I1-3, I1-4, and I1-5). The columns labeled Parent Tag and Child Tag fields on the Foxpro referential integrity builder are the parent and child tables that are related. Note that in most cases, the parent and child tag have the same name. The parent and child tags do not have to share the same name but the tags must be of the same data type and length. The Update, Delete, and Insert columns refer to the actions a user may take when editing a record. Within these columns, the integrity rules, *Cascade*, *Ignore*, and *Restrict* define the action taken when changes made to the parent table are passed to the child table. If *Cascade* is selected, then all changes to the parent table are passed to the child table. If *Ignore* is chosen, then changes made to the parent table are ignored in the child table. If *Restrict* is chosen, then changes to the parent table are restricted if there are related records in the child table.

Data Dictionary

The Data Dictionary (Table 1) describes each data entity and the fields that make up that entity. Each field is described by its type (e.g. character, or numeric), number of characters in the field (field width or length), format of data, and description of information the field is to contain.

Interface with Previous Database Specifications

The original database developed in Phase I of the SRS Environmental Dose Reconstruction Project will be incorporated into the new database.
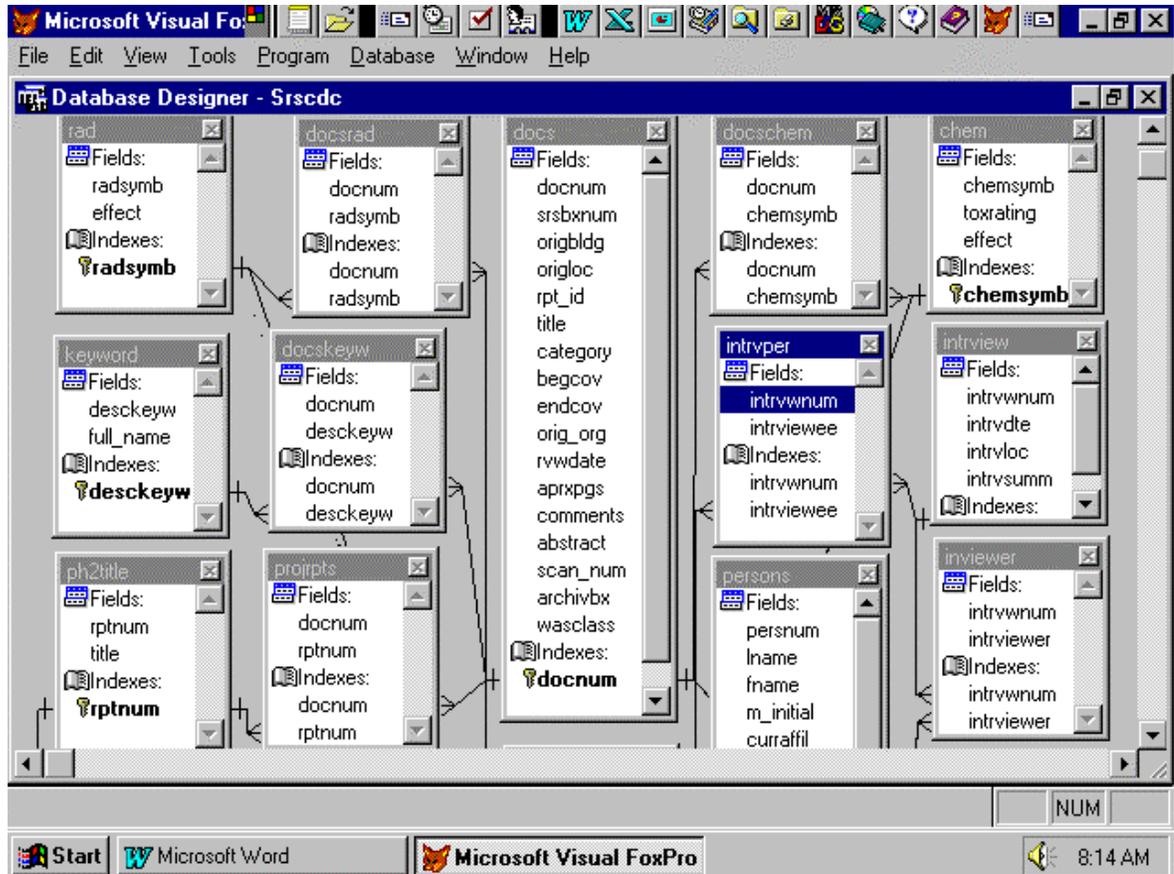
Figure I1-2. Data entity relationship diagram generated by Foxpro. The shaded area represents individual data tables. Within each table are the fields that comprise the table, and indexes used to sort and retrieve field entries.
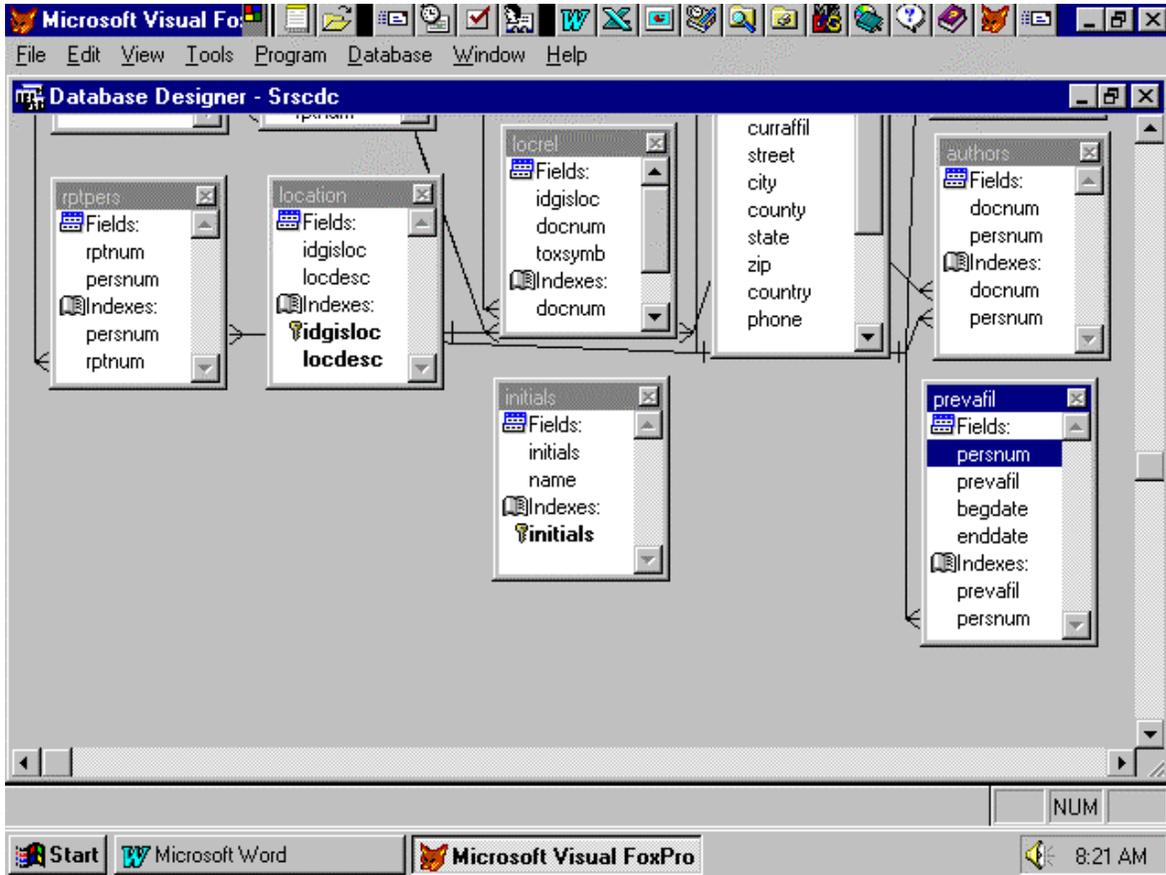
Figure I1-3. Data entity relationship diagram generated by Foxpro. The shaded area represents individual data tables. Within each table are the fields that comprise the table, and indexes used to sort and retrieve field entries.
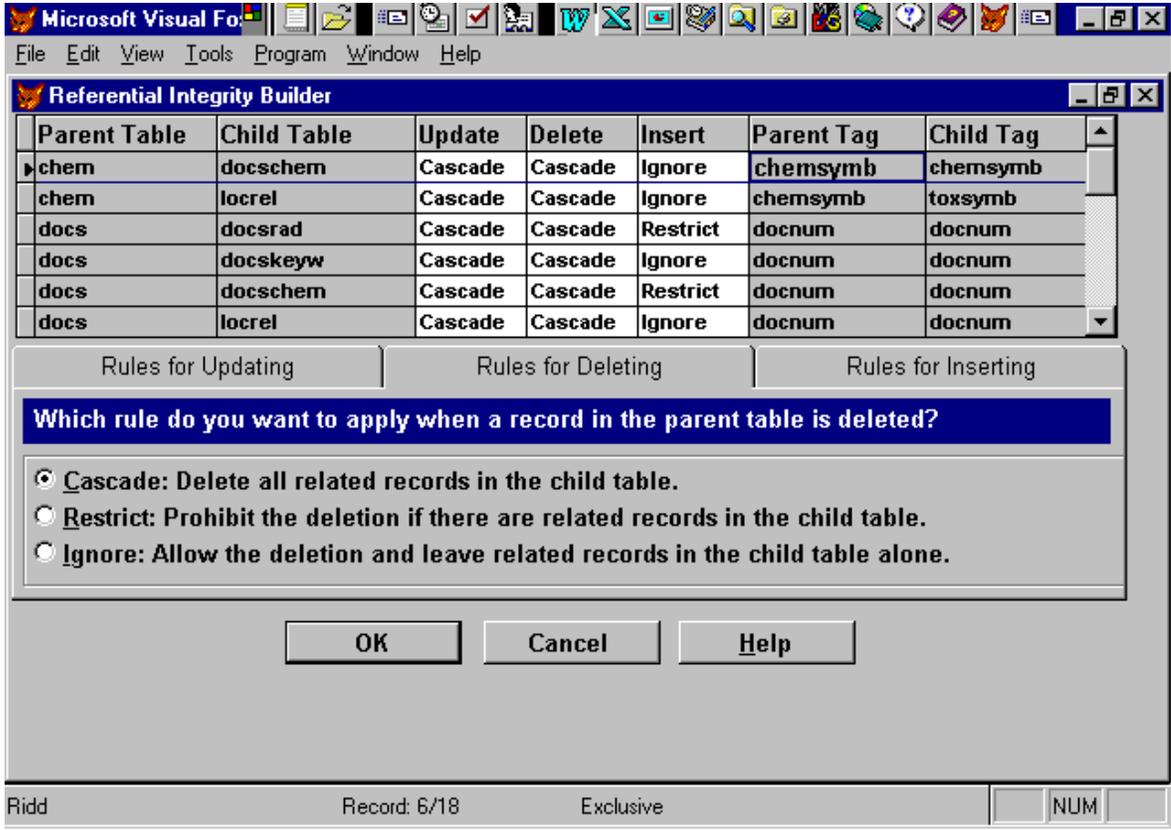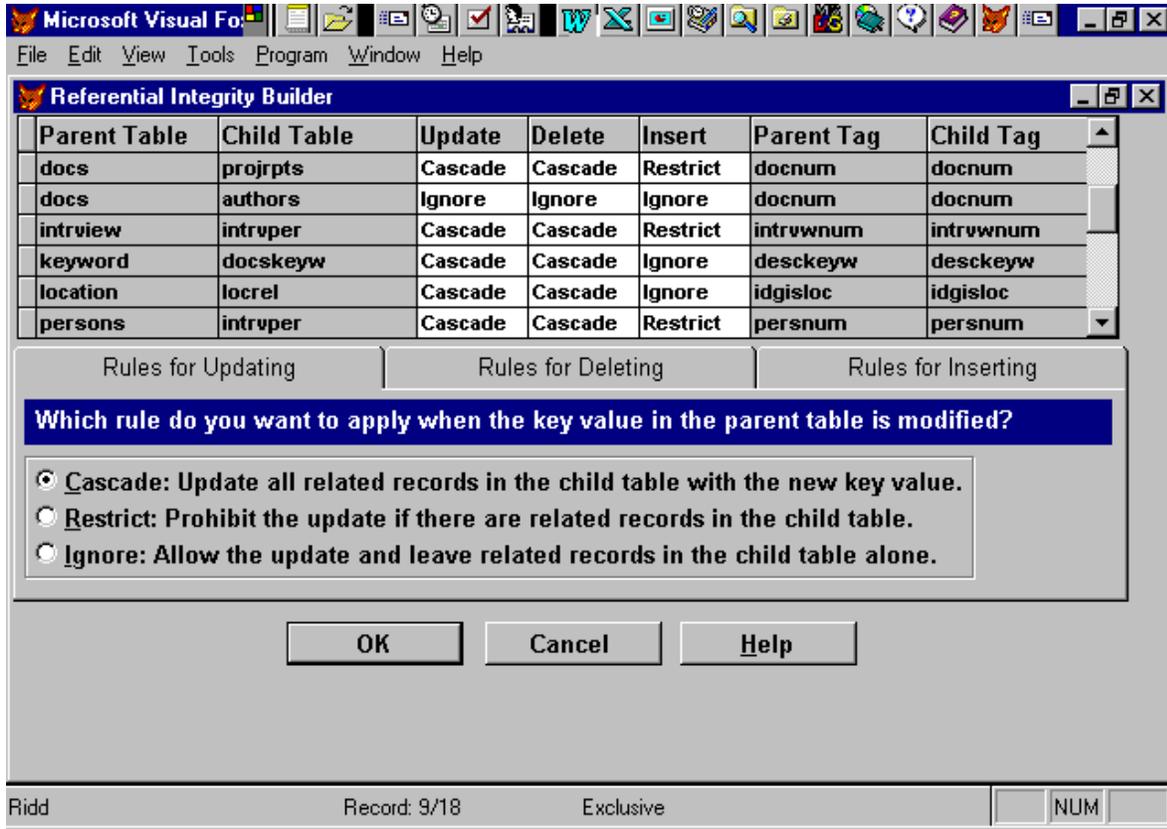
Figure I1-4. Referential integrity for the *chem* and *docs* tables.

Microsoft Visual Fo:

File Edit View Tools Program Window Help

Referential Integrity Builder

	Parent Table	Child Table	Update	Delete	Insert	Parent Tag	Child Tag
	docs	projrpts	Cascade	Cascade	Restrict	docnum	docnum
	docs	authors	Ignore	Ignore	Ignore	docnum	docnum
	intrview	intrvper	Cascade	Cascade	Restrict	intrvwnum	intrvwnum
	keyword	docskeyw	Cascade	Cascade	Ignore	desckeyw	desckeyw
	location	locrel	Cascade	Cascade	Ignore	idgisloc	idgisloc
	persons	intrvper	Cascade	Cascade	Restrict	persnum	persnum

Rules for Updating	Rules for Deleting	Rules for Inserting

Which rule do you want to apply when the key value in the parent table is modified?

⦿ **C**ascade: Update all related records in the child table with the new key value.
○ **R**estrict: Prohibit the update if there are related records in the child table.
○ **I**gnore: Allow the update and leave related records in the child table alone.

OK	Cancel	**H**elp

Ridd   Record: 9/18   Exclusive   NUM

Figure I1-5. Referential integrity for the *docs*, *intrview*, *keyword*, *location*, and *persons* tables.
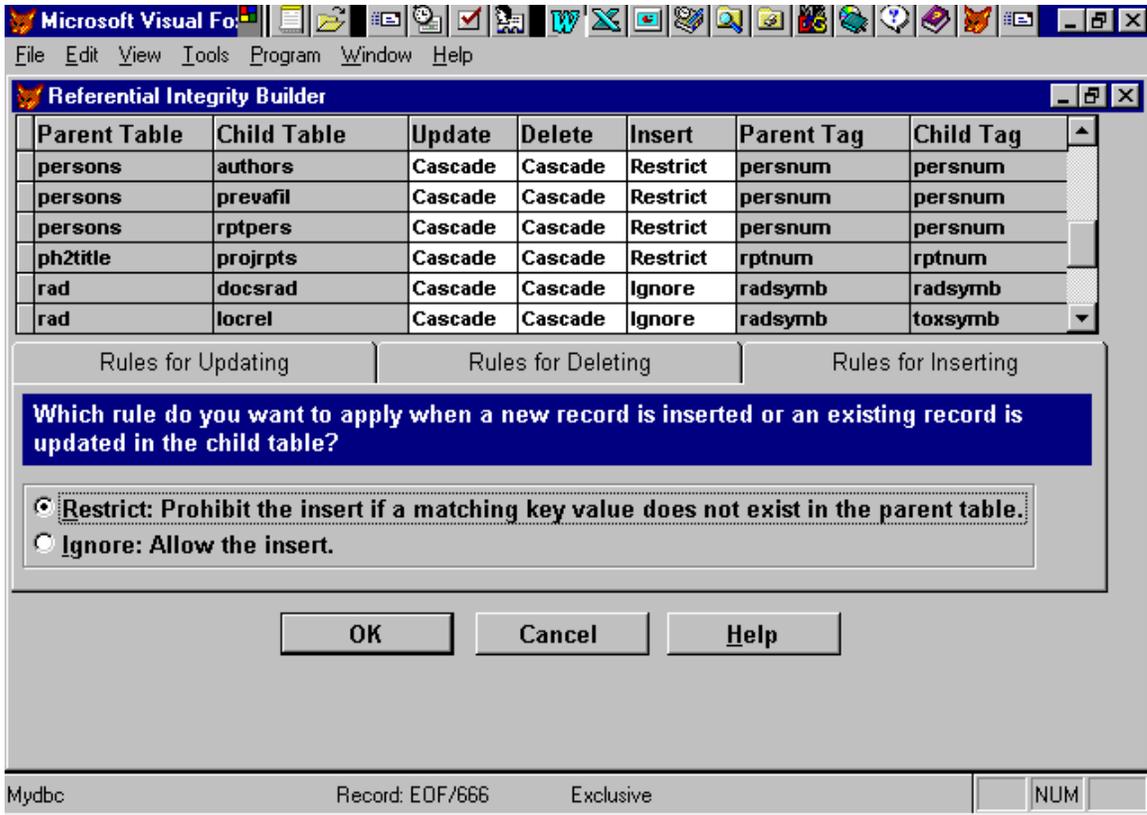
Figure I1-6. Referential integrity for the *persons*, *ph2title*, and *rad* tables.

Table I1-1. SRS Phase II Document Database Data Dictionary

Table or Field Name	Field Name[a]	Field Width	Field Description
DOCS.DBF			
DOCNUM	C	16	A unique string automatically assigned by the database to each document entered. The number is automatically created from: the reviewer's initials; the PC's current date value; and an auto-incremented serial number for each reviewer, reset to zero each day. The format of DOCNUM is: Reviewer initials, year, month, day, and the serial number, e.g. HRM 1993 7 25 1; HRM 1993 7 25 2, etc.
SRSBXNUM	C	20	Boxes stored in the repositories at SRS have been assigned box numbers by SRS records personnel, recorded in this field. Not all records in the database will have a box number, because some were obtained from places other than the official SRS repositories, including file cabinets in staff offices. SRSBXNUM is not useful as a separate table, since SRSBXNUMs are not unique identifiers. Linking documents to non-unique box numbers via a table might create confusion during attempted retrieval of original documents. Also includes identifying characters assigned to records discovered in boxes, file cabinets and other locations offsite.
ORIGLOC	C	20	Original location of SRS document box. The location of an SRS box is subject to repeated change over time. SRS maintains a computerized tracking system capable of locating its records boxes upon request. In the event that Category 0 or 1 records must be relocated during Phase II and later work, the SRS tracking system will be utilized. Category 2 or 3 documents are copied and retained external to the SRS system, in *RAC* archive boxes.
ORIGBLDG	C	15	SRS building in which records box was originally stored.
RPT_ID	C	20	May include document's memo number, report number, publication (journal), Library of Congress ID, or other identification, if applicable.
TITLE	C	254	The official document title. If untitled, researcher creates a title based on document content. A created title is placed in parentheses.
CATEGORY	N	1	Category **2** and **3** documents are likely to be of significant value to the study. Both Category 2 and 3 documents are copied to CDC's contractor and the Aiken reading room. Category **3** documents have been determined by the discovering researcher to be of sufficient potential value to the dose reconstruction that they should be abstracted into the database. Category **1** records are of potential value, but are likely to be available in more original form elsewhere. Category **0** includes documents of no value to dose reconstruction.

Table or Field Name	Field Name[a]	Field Width	Field Description
			Reference documents used during the Phase II research may be copyright-protected and not copied to the Project's archive boxes. They are defined in this database as Category 1 records. CATEGORY is a required entry field.
BEGCOV	D	8	The document's coverage beginning date, identified during inspection of the document.
ENDCOV	D	8	The document's coverage ending date, identified during inspection of the document.
ORIG_ORG	C	15	Organization that produced the document
RVWDATE	D	8	The date that the Dose Reconstruction Project researcher first encountered the document and initiated its entry into the database. Not related to the original date of the document creation, or to the date abstracted by an assistant, or to later reviews by other researchers.
APRX_PGS	N	5	Approximate page count of the document. The default value for this field is 1. Large documents were often assigned the value 1000, to indicate significant copying effort, during Phase I research.
COMMENTS	M	10	Contains previous contents of Phase I SRSDOSE Comments memo fields; contains comments from Phase II research which don't fit logically elsewhere in the database.
ABSTRACT	M	10	Contains abstract if document is Category 3
SCAN_NUM	N	12	If some documents are eventually scanned and made available in electronic format, this will serve as a link to the scanned data.
ARCHIVBX	C	15	All copied documents are in Project possession offsite. ARCHIVBX indicates the specific *RAC* box in which the copied Category 2 or 3 document is stored. This ID also points to the specific microfilm cassette containing the document.
WASCLASS	L	1	Indicates whether the document was originally classified (Y) or not (N). All documents used during dose assessment will have been declassified. A document's original classification level (CONFIDENTIAL, SECRET, or TOP SECRET), or special caveats (NOFORN, RESTRICTED DATA, FORMERLY RESTRICTED DATA, NAVAL NUCLEAR PROPULSION INFORMATION, etc.) is not generally noted in the database. Default is N.
INITIALS	C	3	Initials of document reviewer

PROJRPTS.DBF			(SRS Dose Reconstruction Project Reports)
DOCNUM	C	14	(See DOCS.DBF)

Table or Field Name	Field Name[a]	Field Width	Field Description
RPTNUM	C	15	Number assigned to chapter or appendix
PH2TITLE.DBF			
RPTNUM	C	15	(See PROJRPTS.DBF)
TITLE	C	254	Title of chapter or appendix
RPTPERS.DBF			
RPTNUM	C	15	(See PROJRPTS.DBF)
PERSNUM	C	10	(See PERSONS.DBF) As used in PROJRPTS.DBF, these are the authors of Phase II project reports, only.
INTRVPER.DBF			(Persons interviewed or doing interview)
PERSNUM	C	10	(See PERSONS.DBF)
INTRVWNUM	C	10	(See INTRVW.DBF
INTRVIEWEE	L	1	Yes or no (is the person an interviewee?)
INTRVIEWER	L	1	No or yes (is the person an interviewer?)
INTRVW.DBF			
INTRVWNUM	C	10	Unique ID number
INTRVDTE	D	8	Date interview held.
INTRVLOC	C	30	Location of interview
INTRVSUMM	M	10	Summary of interview.
DOCSKEYW.DBF			
DOCNUM	C	16	(See DOCS.DBF)
DESCKEYW	C	20	(See KEYWORD.DBF)
KEYWORD.DBF			
DESCKEYW	C	20	Word or phrase to quickly characterize document content
FULL_NAME	C	40	Definition of descriptive keyword
DOCSCHEM.DBF			
DOCNUM	C	16	(See DOCS.DBF)
CHEMSYMB	C	30	Unique symbol identifying toxic agent
CHEM.DBF			
CHEMSYMB	C	30	Unique chemical ID (e.g. CCl4)
TOXRATING	C	50	Toxin's hazard rating
EFFECT	M	10	Summary of potential effects of exposure to toxin
AUTHORS.DBF			
DOCNUM	C	16	(See DOCS.DBF)

Table or Field Name	Field Name[a]	Field Width	Field Description
PERSNUM	C	10	(See PERSONS.DBF)
PERSONS.DBF			
PERSNUM	C	10	Unique ID
LNAME	C	15	Last name
FNAME	C	15	First name
M_INITIAL	C	2	Middle initial
CURRAFFIL	C	30	Current affiliation
STREET	C	30	Mailing address
CITY	C	15	
COUNTY	C	15	
STATE	C	2	
ZIP	C	10	
COUNTRY	C	15	Defaults to "U.S.A."
PHONE	C	12	
PREVAFIL.DBF			
PERSNUM	C	10	Unique ID
PREVAFIL	C	30	Previous affiliation of individual
BEGDATE	D	8	Date previous affiliation began
ENDDATE	D	8	Date previous affiliation ended
DOCSRAD.DBF			
DOCNUM	C	16	(See DOCS.DBF)
RADSYMB	C	30	(See RAD.DBF)
RAD.DBF			
RADSYMB	C	30	Unique ID character set for radionuclide (e.g., Pu-238)
EFFECT	M	10	Discussion of radionuclide's principal characteristics
LOCREL.DBF			(Main locations for toxic releases: buildings., facilities, etc.)[b]
LOCDESC	C	30	Unique reference number
DOCNUM	C	16	(See DOCS.DBF)
TOXSYMB	C	30	Radioactive material or toxic chemical symbol
LOCATION.DBF			
LOCDESC	C	30	Unique reference number
LOCDESCR	C	30	Brief descriptions of possible release locations. Used as a lookup table.

Table or Field Name	Field Name[a]	Field Width	Field Description

[a] Field Type: C = character, N = numeric, D = date, M = memo, L = logical. Mandatory data entry field if the character in this column is boldfaced.

[b] Detailed location data (e.g., sample and, sampler latitude/longitude) are maintained in the Phase II Geographic Information System database.

REFERENCE

Radiological Assessments Corporation (*RAC*), 1995. *Technical Proposal: Source Term Calculations and Ingestion Pathway Data Retrieval for the Savannah River Site (SRS) Environmental Dose Reconstruction Project*., May 11, 1995.

APPENDIX I-2

SOFTWARE TEST PLAN FOR THE DOCUMENT DATABASE

APPENDIX I-2

SOFTWARE TEST PLAN FOR THE DOCUMENT DATABASE

PURPOSE AND BACKGROUND

This document provides the testing plan to verify and validate the Document Database software as described in Task 3 of the *Radiological Assessments Corporation* Technical Proposal for Source Term Calculation and Ingestion Pathway Data Retrieval for the Savannah River Site (SRS) Environmental Dose Reconstruction Project, May 11, 1995 (RAC 1995). Verification is defined as the process of determining whether or not the products of a given phase of the software development cycle fulfill the requirements established during the previous phase. Validation is defined as the process of evaluating the software at the end of the software development process to ensure compliance with the software requirements.In this project, individual phases for developing of the Document Database were not clearly delineated. Therefore, this test plan covers verifying of each of the code modules and validating that the code as a whole fulfills the requirements stated in the software requirements and design specifications document. This document is intended for review by personnel at the Centers for Disease Control and Prevention (CDC) and software developers under subcontract with the *Radiological Assessment Corporation.*

The Document Database was developed for the SRS Environmental Dose Reconstruction Project to track documents, which are defined as any material that provides information or data used in the project. The database tracks

- Published internal documents and reports,
- Journal articles and books,
- Unpublished reports, memos, and letters,
- Field or office notebooks and log entries,
- Laboratory analysis reports and quality assurance documentation,
- Databases, electronic or otherwise, containing information used in the study (e.g. monitoring data,
- Documented information and data obtained from interviews or phone conversations, and
- other documented information.

The software will use the Foxpro$^{©}$ relational database software and programming language. Foxpro databases, or tables (designated with a .DBF file extension), will be created for each *data entity*. Each entity may contain numerous data *fields*. A field contains the information that is either entered or retrieved by the user. A field is designated as a character, numeric, date, logical, or memo field. Character, date, memo, and numeric fields have lengths defined by the developer. That is, the field has a limited number of characters that may be entered in the field by the end user. Logical fields have a predefined width of 1. Tables are usually linked to other tables by fields of the same name and data type. In this way, the information presented to the end user may originate from several tables that are related by fields of the same name and type.

The Foxpro programming language will be used to create a user interface to enter or query the database. The program will be compiled into an executable file for distribution to end users.

OBJECTIVES

This test plan for the Document Database will accomplish the following objectives:
- Detail the activities required to prepare for and conduct the software test,
- Communicate to all responsible parties the tasks that they are to perform and the schedule to be followed when performing the tasks,
- Define the sources of information used to prepare the plan, and
- Define the operating environment needed to conduct the test.

SCOPE

This test plan covers the testing of the Document Database software by independent reviewers. The developers may also use the procedures defined in this document for internal testing before release. The testing will cover referential integrity of the tables and the application software designed to manage and control input, output, and editing of the tables.

TEST ITEMS

The testing of a software product should include four components. These are
1. Functional coverage,
2. Topological coverage,
3. Component interface,
4. Performance measures.

Functional coverage refers to how many of the code processes and data (if applicable) are exercised during the testing. An example of a process is the editing and retrieval features built into the Document Database. Data refers to the pieces of information stored, retrieved, and manipulated by the software.

Topological coverage refers to the logic path that the code takes to determines if a particular function, task, or feature is executed. For example, the logic path the code may take to allow record retrieval may involve first checking the user identification and then checking if the record exists.

Component interface refers to how well each of the code components or modules work together as a whole. Each module or component may work fine independently. Testing the component interface verifies that the code logic linking each component operates correctly.

Performance measures refer to measures that quantitatively describe how well the code performed during testing.

In this test plan, we have attempted to address all four of these components by designating a test database that will include entries in all data entities and data fields, and testing all code features that have been written and implemented. Neither functional or topological coverage is 100% assured, but through the methods outlined in this test plan, we can be reasonably assured that most situations a user may encounter during normal operation of the code are tested.

The software tested will be designated the Beta Test Version until successful completion of the testing procedures.

FEATURES TO BE TESTED

User interface application modules are written under the Visual Foxpro object-oriented programming language. The user is presented a series of menu items that perform specific tasks. These items have methods and properties attached to them and it is these features that will be tested, along with the relational integrity of the data tables that are linked to their operation. A main menu is presented first and includes the items typically found on most Windows -based applications. Under each main menu item is a series of submenus that exercises different options, depending on the needs of the user. The main menu items shown in Figure I2-1 will be tested. The referential integrity of each table will also be tested as stated in the software requirements and design specifications document (Rood et al. 1996).

APPROACH

A test database will be used that will include entries for all data tables and lookup tables. The test database will be relatively small; therefore, output from the tables can be easily checked against the data in the test database. The test database will be developed before beginning any testing. The hardcopy printout of the test database will be used to measure software performance.

A person designated as the test technician will perform the actual software testing. The test technician will perform tasks such as data entry, retrieval, and report generation. Results from each test will be provided to the test analyst who will determine if the test was performed correctly and if the results are satisfactory. The test analyst will provide written documentation that the test was completed satisfactorily. This documentation may take the form of a letter report. The test technician and test analyst will be independent of the development team. We suggest the CDC staff perform these duties. The developer may use the testing procedure outlined in this document to test the software before release, but is not required to provide these results to the CDC. Referential integrity diagrams, data dictionary, and database design diagrams will be provided to the test analyst before testing begins. These items are described in Rood et al. (1996). The database design diagram consists of a listing of each of the tables contained within the database application. Under each table, the fields associated with that table and indexes are listed. The referential integrity diagrams define how changes made to one table are passed to other tables.
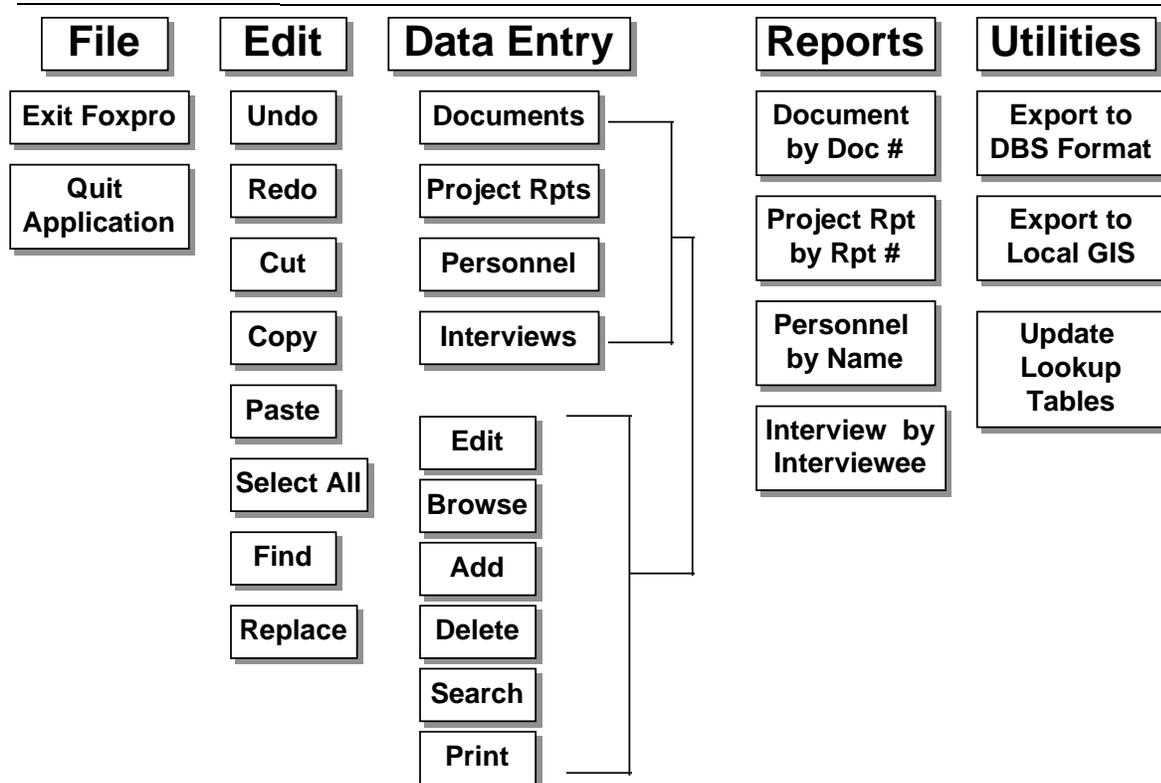
Figure I2-1. User interface menus for the Document Database. The GIS designation under the utilities menu refers to a Geographical Information System.

Testing will be performed on the Document Database software identified as the Beta Test Version 1.0. The testing procedure will be composed of three phases. In the first phase, data entry will be tested, in the second phase, document query, additions, deletions and editing will be tested, in the third phase, report generation, utilities, lookup tables, and housekeeping options will be tested. Each phase is discussed below.

Phase 1 Testing: Data Entry

Phase 1 testing will exercise the applications written to enter data from the keyboard into the various data tables. Menu items exercised in this phase will include the Add sub-menu under Data Entry in the main menu. It will also test the entity relationships between data tables. The test database will be entered manually and reviewed for accuracy by the test technician. A hardcopy printout of the database will be provided to the test analyst who will determine if the data present on the hardcopy printout is indeed in the database and that it was placed in the proper table. A query of each individual table performed in the interactive mode of Foxpro and outside the query applications shall be performed in order to verify the information presented on the hardcopy is present in the table or tables designated to store that information.

Phase 2 Testing: Document Query and Editing

Phase 2 testing will exercise the Documents, Project Reports, Personnel, and Interviews sub-menu options under Data Entry in the main menu. In each of the sub-menus, the search, delete, edit, and print options will be exercised. A subset of the test database will be selected for searches, employing the user interface to perform the searches. The subset of the test database will be selected by the test analyst. The subset will include entries from each of the data categories (Documents, Project Reports, Personnel, and Interviews). The test technician will query the database for each record in the subset and print the results. The hardcopy printout will be provided to the test analyst to verify each record was retrieved and printed properly. The subset of the entries will be selected for deletion by the test technician. The test analyst will then query the database for each record in the subset to verify that the records were deleted. These records will then be added back by the test technician.

Each record in the subset of the test database will be selected for editing. Each record will be retrieved and edited using the user interface by the test technician. The test technician will provide the test analyst a record of all editing changes made to the database. The records will then be retrieved by the test analyst and verified to make sure the changes were saved and that referential integrity of the table is intact. That is, the updates, deletions, and inserts are passed from the parent table to the child table correctly.

Phase 3 Testing: Report Generation, Utilities, Lookup Tables, and Housekeeping

Phase 3 testing will exercise the report generation options of the user interface. The test technician will generate reports for the entire test database by document number, project report number, personnel, and interviewee. The reports will be reviewed by the test analyst and visually compared with the hardcopy printout of the database to assure all records that should have been printed were printed.

The test technician will then export the entire test database to both DBS format and local Geographic Information System (GIS) format. The DBS-formatted export file will then be imported into another database application by the test technician and provided to the test analyst. The test analyst will then check each record in the other database application to assure records were exported from the Document Database correctly. The GIS-formatted export file will then be imported into a local GIS application by the test technician and provided to the test analyst. The test analyst will then check the GIS database to verify each record was exported correctly from the document database.

PERFORMANCE MEASURES

Performance measures used during testing will be pass, fail, and conditional pass. Each item described in the three test phases previously described will be assigned one of these measures. A pass will be assigned to a test if items selected for editing, retrieval, printing, or export match *exactly* what is present in the test database hardcopy printout. A fail will be assigned to a test if either the software fails to operate or the software generates results that are inconsistent with the hardcopy printout of the test database. A conditional pass may be assigned to a test if the software functioned properly during the test, but the style and format of the results are undesirable.

REPORTING

The test analyst should document the test results and provide any pertinant information (such as deficiencies in performance), either verbally or in writting, to the software developer. Documentation should include all test results and any abnormalities noted by the test analyst. This documentation should be saved by the test analyst. If during the testing procedure a portion of the code does not operate correctly and makes continued testing impossible, then the test analyst will notify the software developer of the problem immediately. The software developer will make the necessary corrections and re-issue a new version of the software. The new software will be designated Beta Test Version 1.1 for the first occurrence and 1.2, 1.3 etc. for any future occurrences. All testing done prior to the occurrence will be redone using the new the software to assure corrections did not impact other subroutines in the software.

The test report should not include recommendations for improvements to the code. However, these may be requested separately. The report should include results from each of the tests identified in this document. All tests should be performed using the same Beta Test Version number.

After the software developer receives verbal test results, appropriate corrections to the software will be made, and the software will then be released to the CDC, identifying it as Version 1.0.

OPERATING ENVIRONMENT FOR TESTING

The software will use the Foxpro relational database language and will require a microcomputer operating under MS DOS$^{©}$, Version 6.0 or higher, Windows Version 3.1 or higher, and at least 8MB of extended random access memory although 16MB is preferable. About 200MB of free hard disk space will be required to load and operate the software. A copy of Visual Foxpro should be loaded on the computer performing the testing.

REFERENCES

Radiological Assessments Corporation (RAC), 1995. *Technical Proposal: Source term Calculations and Ingestion Pathway Data Retrieval for the Savannah River Site (SRS) Environmental Dose Reconstruction Project.* Neeses, South Carolina, May 11, 1995

Rood, A. S., M. O. Langan, H. R. Meyer, 1996. *Software Requirements and Design Specifications for the Document Data Base. Radiological Assessments Corporation* DRAFT Technical Memorandum, June19-CDH-SRS-1996.