AN IMPLEMENTATION OF THE FACETED CLASSIFICATION SYSTEM FOR SOFTWARE REUSE

By

RAM R. MAREDDY

Bachelor of Technology

Jawaharlal Nehru Technological University

Hyderabad, India

1988

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
December 1994

AN IMPLEMENTATION OF THE FACETED CLASSIFICATION SYSTEM FOR SOFTWARE REUSE

Thesis Approved:

Mansur Samadzadeh

Thesis Advisor

Blayne E. Mayfield

H. Lu

Thomas C. Collins

Dean of the Graduate College

ACKNOWLEDGMENTS

I take this opportunity to express my gratitude to Dr. Mansur Samazadeh for his guidance and encouragement. Though my thesis work took a lot more time than the usual, Dr. Samadzadeh gave me continuous moral support. I appreciate him greatly for this personal and intellectual relationship. I also like to thank Drs. Mayfield and Lu for agreeing to be on my thesis committee, and for their comments and advice.

I also like to thank my family and friends for their friendship and support.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Reusable software is widely believed to be the key to increased productivity in software development. While every "new" software system more often than not requires several components that have been developed before, these components frequently are re-developed for the current system. Developers will be able to concentrate on developing truly new components if they could easily integrate past developments into their current one. Even though the strategy of reusability offers great promise, it has been generally unfulfilled to a great extent [Biggerstaff87] [Hall92].

The beginnings of software reuse can be traced back to the 1968 NATO Software Engineering Conference which focused on building large, reliable software systems in a cost effective way [Krueger92]. Recent renewed interest in software reuse has a lot to do with the cost of developing software. With the cost of powerful hardware falling everyday, ambitious and more powerful software systems are being planned to provide solutions for various practical problems. Quality along with productivity will improve in building these systems, if well-tested and already developed components were to be used in constructing these systems.

One of the fundamental problems in reusing software is the storage, search strategy, and retrieval of the components that are required for building systems. Whenever a component is built, a library at a central location should be available for keeping this component for retrieval at a later time. For both these purposes, a classification system should be available which, when incorporated into a tool, will facilitate the storage and retrieval of software components.

As the main objective of this thesis, a tool was developed to help in reusing the existing software components. This tool uses the faceted classification system in organizing the software components.

Chapter II of this thesis reviews the current literature on software reuse. It also defines some of the basic terms used in software reuse, details the past and present efforts in software reuse, and also provides the basic background on software reuse. Part of this literature review discusses the faceted classification scheme, which is the basis for the tool developed as a part of the thesis. Since the database requirements are basically relational, the theory and description of the relational design is also briefly described in this chapter. Chapters III and IV discuss the tool in detail. Details about the implementation platform, design and implementation of this tool, and a sample reuse case are presented in these two chapters. Chapter VII summarizes the effort and briefly mentions the scope of future work in this area. Appendix E has user and system administrator guide that explains how to use this tool. All functionality is clearly explained in this guide. All the source code is attached as the Appendix F.

CHAPTER II

LITERATURE REVIEW

2.1 Definitions

This section contains some basic definitions of software reuse that are used in this thesis. Most of these definitions are taken from two main references [Krueger92] [Prieto-Diaz91].

Software reuse can be defined as using the existing software artifacts in building a new software system. A software artifact can be a source code fragment, a design structure, a module-level implementation structure, a specification, a documentation item, a transformation, etc.

Abstraction refers to hiding the implementation details of source code and providing only the necessary information about the functionality of that source code. Abstraction is an important feature of software reuse. Without abstraction, the reuser will be forced to spend a lot of time trying to understand what each artifact is doing and how it can be reused.

Selection refers to choosing a particular software artifact from a collection according to the needs of the reuser. If a classification system is used to organize the collection, the selection process will be greatly simplified.

Specialization is transforming an existing software artifact to meet the current needs. This specialization is achieved through parameters, constraints, and other forms of refinement.

Integration refers to adding an existing software artifact to the system that is currently being developed. Module interconnection languages are examples of integration frameworks.

The faceted classification system is a scheme for organization based on facets or attributes of objects. A facet list is a collection of terms that can identify or describe various parts of a facet of the domain. A facet list can also be referred to as the vocabulary for classifying software artifacts.

The relational model refers to organizing data elements through tables and tables only. With the help of common columns amongst the tables, relationships are defined among different tables.

2.2 Software Reuse

2.2.1 History and Past Research Efforts

As mentioned in the introduction, a historically important conference in the field of software reuse was the 1968 NATO Software Engineering Conference that was held in Garmisch, Germany. But the origins of software reuse can be traced back to as early as 1949 when the University of Cambridge proposed the first subroutine library on its stored program computer EDSAC [Tracz88a].

A more recent event in the field of reuse is applying it to alleviate the "Software Crisis" in the context of the development of the Ada programming language. This language was developed under the auspices of the US Department of Defense (DoD) in the early 80's. Software reuse and the object-oriented approach are two of the aspects of program development that Ada's features were expected to support [Tracz87].

Ada has several language constructs to facilitate the development of reusable components. These include a package construct, which separates specification and implementation, and an overload resolution construct, which facilitates the semantic and syntactic reuse of functions. Ada also enforces strong typing. The Department of Defense

directives on standardization, validation, and mandated use of Ada has enhanced the chances of software reuse making a significant impact on software crisis at DoD.

A large number of corporations also have contributed to the field of software reuse. These corporations include Boeing, Ford, General Dynamics, IBM, Lockheed, and Honeywell. They have projects addressing several aspects of reuse such as standardization of support tools, construction of libraries, automated catalog and library interaction, and coding without language or implementation tricks [Tracz87].

2.2.2 Current Practice

As discussed in the previous section, the concept of software reuse has been in practice for some time. The following subsections detail the different ways this concept is practiced [Krueger92] [Hall87] [Zand94] [Zand93] [Zand92] [Swanson92].

2.2.2.1 Vertical Reuse or High-Level Languages. Reuse in languages started with assembly languages. Assembly language routines provided an abstraction to the machine language routines for most of the hardware-level functions. In the newer languages, libraries of standard functions further extend this abstraction. Object-oriented languages such as C++ and Smalltalk provide powerful features for the user to extend such abstractions. When the concept of abstraction is further extended, we get Very High-Level Languages (VHLL). These languages resemble application generators in the sense that the specification for a certain task is automatically transformed into executable systems. This property derives another name for VHLLs - executable specification languages.

2.2.2.2 Horizontal Reuse. Within a level of abstraction, functions can be reused. Pipes and filters of UNIX facilitate reuse under this category. In this case, the output of one function is transferred to another function; in effect reusing the second function without having to rewrite it. Batch processing also comes under this category. Horizontal reuse is

different from vertical reuse in the sense that the first category uses high-level languages whereas the second category uses the operating system functionality.

2.2.2.3 Design and Code Scavenging. Sometimes programmers scavenge code and designs from a previously developed systems. Typically, the locations of this type of artifact and the concepts used reside primarily in the programmers' head. So, recalling from memory, a programmer will copy code fragments from an old system and integrate them into the current project. This type of reuse becomes more effective as the programmer becomes more experienced.

2.2.2.4 Source Code Components. Similar to the hardware component industry, this notion suggests an industry of off-the-shelf source code components. Ideally, functions with clearly defined input and output values can be manufactured for use in several different programs. The packaging of these components can in principle use systematic techniques such as catalogs and libraries of components. To use these functions, a reuser can set the parameters rather than editing the source code directly, even with the availability of source code in such component libraries.

2.2.2.5 Application Generators. For application generators, the input is the specification of the required task and the output is the executable program that addresses the required task. These are similar to conventional programming language compilers, except that they are at a higher level. Application generators typically focus on a narrow domain and generate code to solve the problems in that area only. Application generators work at such a high level that they concentrate on what the system should do rather than how it is done. So, algorithms and data structures are automatically generated for the reuser. Application generators have been developed in several domains including database management, textual report generation, and graphical report generation [Burton87].

Parsers and compiler generators like lex and yacc come under this category. Lex addresses lexical analysis and yacc deals with parsing.

2.2.2.6 Software Schemas. In software schemas, the notion is reusing algorithms and data structures rather than the code itself. An example of this schema is the PARIS system [Katz87]. In this system, a reuser starts the development by giving a problem statement, which is a formal set of computational requirements. Then, with its sophisticated retrieval system, PARIS supplies the schema that satisfies the problem statement.

2.2.3 Reuse Classification Criteria

Reusability has three important aspects: the abstraction level, customization methods, and reusability conditions [Lenz87] [Krueger92]. A classification system used for software should facilitate all these three aspects of reuse.

2.2.3.1 Abstraction Level. Abstraction is the most important feature of software reuse. If the implementation details of the software artifact are not hidden, a programmer will be forced to spend more time in understanding the internals. The comprehension overhead may eventually force the programmer to redevelop that software artifact instead of reusing it. A reasonable amount of abstraction should be practiced in all units of software, be they specifications, designs, or code.

2.2.3.2 Customization Methods. Once a software component is identified as a candidate to be used in a new system, it has to be tailored to meet the current needs. This can be done by simple parameterization or by changing of the internals. When a candidate artifact needs no tailoring at all, obviously we have the best form of software reuse.

2.2.3.3 Reusability Conditions. When the domain, in which the software artifacts are being developed, reaches a certain degree of maturity, concepts used in that domain become apparent. This paves the way for higher software reuse. In a new domain,

reusability becomes difficult because of the lack of software artifacts and the lack of domain knowledge by the programmer. So, a thorough analysis of the domain will help improve software reuse.

2.2.4 Economics of Reuse

As mentioned in the introduction, the chief reason behind the increasing interest in software reuse is the enormous increase in the cost of developing complex software. A simple cost model proposed by John Gaffney of the Software Productivity Consortium is as follows [Tracz88b] [Barnes87] :

$$C = (1-R)*1 + b*R$$

where C is cost of developing software, R is the percentage of code reused, and b is the cost of reusing a line of code / cost to develop a new line of code.

As we can see from the above model, savings will be more when more code is reused or when the development of new code is higher than reusing the existing code. But all these gains, that can be realized from the reuse of software, are moderated by some organizational factors. Project managers in large organizations are not typically rewarded for the reuse of old software. Without organizational commitment, it becomes difficult to develop and maintain libraries with reuse in mind [Barnes87].

2.3 Faceted Classification System

An important aspect of software reuse is the organization of software artifacts so that they can be searched and retrieved easily. This organization methodology also should provide an easy way of inserting newly developed artifacts into the collection.

Three concepts of organization are widely used throughout conventional libraries in the world. The first one is the Dewey Decimal System, which is used in a number of libraries in the United States, the second one is the Library of Congress Classification System, which is used by the Library of Congress and several other libraries in the United

States, and the third one is the Faceted Classification System, which is widely used in Europe and India [Prieto-Diaz91]. In the Dewey Classification System, the possible classes are predefined. Hence, when a title needs to be classified under this system, the librarian should find a class that best fits the title. In order to do this, the librarian should have expertise in both the Dewey Classification System and the subject matter that the title represents. As expected, with several closely related classes, a title spanning across several classes becomes difficult to classify.

The Library of Congress (LC) System is similar to the Dewey Classification System in many respects, especially in terms of the predefined classes. But its notation differs significantly from Dewey. The following example, from [Immorth71], explains the difference. The third edition of Richard D. Altick and Andrew Wright's *Selective Bibliography for the Study of English and American Literature* has a call number Z2011.A4 1967 under the Library of Congress System, and 016.82 A468s 1967 under the Dewey Decimal system. Each of these call numbers has three components: Class Number (Z2011 in LC and 016.82 in Dewey), Author Number (A4 in LC and A468s in Dewey), and Publication Date (1967). Z2011 and 016.82 are predefined classes in both of these systems.

With the Faceted Classification System, it becomes easy to decide on titles spanning across several classes. A faceted scheme contains as many facet lists as the domain to be classified requires. Each facet list contains as many keywords as needed to describe that facet. So, when a librarian wants to classify a particular title under this system, the librarian will select a term from each facet that suits the title. After all the terms are selected, the class that is arrived at will be the best fit for this title, hence offering flexibility and accuracy.

Another recent approach to information retrieval is by free text analysis [Prieto-Diaz91]. There are several reasons why free text analysis does not work for the analysis of source code functions. First of all, there might not be a lot of free text in the source

code - unless the programmer decides to do a lot of commenting. Variable naming conventions can differ from programmer to programmer and also from function to function. In effect, by free text analysis of a function, it is not in general possible to find out what a function is doing or how it is doing it. In contrast, in the Faceted Classification System, there are predefined lists of key words. This list of key words or standard vocabulary can be updated as needed. Hence this classification not only provides the flexibility of having a reasonable number of key words, but it also provides consistency among the programmers or reusers and the librarians of a reuse system [Prieto-Diaz91].

2.3.1 Facets and Vocabulary

Figure D-3 contains the list of facets and vocabulary that is developed for the PC applications domain. For the purpose of the Faceted Classification System, a total of seven facets are suggested to characterize this domain (this system provides the flexibility of addition, deletion, and modification to these facets at any time). These seven facets are the Operating System for which the software artifact works, the Language in which the artifact is developed, the part of the system where this artifact works best, the action or service this artifact provides, a name given to this artifact by the developer, the name of the author, and the date of creation.

Each of these facets has a list of choices (or vocabulary) to further describe the software artifact. This vocabulary can also be modified as the repository of artifacts grows. The following paragraphs briefly describe the vocabulary for each of these facets.

In a large corporation, normally several operating systems are used on the micro-computer domain. They include DOS, Windows, Windows NT, OS/2, and Unix. Others may be added as they become available. Similarly, several languages might be used in the development process, they include C/C++, Pascal, and FORTRAN.

In the case of operating systems and languages, versions also play an important role. For example, DOS can be further refined into DOS3, DOS4, DOS5, and DOS6. The facets, vocabulary, and files related to the older artifacts can be deleted to make room for the newer ones.

The two facets refined to as system component and action describe the functionality of the artifact. The system component suggests the place where this artifact might be used. The action suggests further specialization of the artifact that is under consideration.

The last three facets deal with the creation of the artifact. The Name given to it by the author, the name of the author, and the date of creation make up this set. Since these are important to identify the artifact, they are also stored in the artifact table of the database.

2.4 Relational Database Management

The relational model refers to a database system that contains tables and nothing but tables at the logical level for organizing data [Date91]. There might also be relations among these tables to connect the data among several tables. So, a user conceptualizes the whole data as tables and relations. Each table contains rows and columns. Each row, or a record from a table, is the complete description of one entity. This entity can be a person, a sale, etc. Each entity is described by several fields, characteristics, or attributes. These fields are represented by columns [MS-SQL93].

SQL (Structured Query Language) is the standard language (with various dialects) that implements the relational database model. This high-level language was originally developed at IBM in the mid-1970s. This language includes statements not only for querying and retrieving data from a database, but also for creating new databases, and modifying and updating them. The American National Standards Institute (ANSI) recently came up with a standard SQL language called Transact-SQL [Transact-SQL93].

The relational model consists of three phases. The first phase in the relational model is data organization. In this phase, the specification of the system is collected in terms of tables, i.e., the rows and columns in each table. Here the primary and foreign keys for each table are also defined. A primary key is one or more columns that uniquely identify the rows or records in the table. A foreign key is a column in a table that is similar to a column in another table. With the help of foreign keys, two or more tables can be joined. The second phase is data definition. In this phase, storage is allocated to the database and then tables are created. So, this phase mainly creates the holders for data. The third phase is data manipulation. In this phase, the functions required for data retrieval and data modification are developed. In the SQL language, data retrieval is primarily done by the SELECT statement, and data modification is done by INSERT, UPDATE, and DELETE statements. As the meanings suggest, to insert new data values into the tables, the INSERT statement is used, to modify the currently existing data, the UPDATE command is used, and to delete any data from the database, the DELETE statement is used.

The above three phases constitute the basic database operations. There are also some additional facilities provided for easy access to data and to keep the integrity of the data [Transact-SQL93]. Indexes can be defined on the tables to access data quickly. There are basically four types of indexes: composite, unique, non-clustered, and clustered [MS-SQL93]. A composite index is created on single or multiple columns. A unique index is created on single or multiple columns, where these columns make a unique key. When a non-clustered index is created, the data is not ordered physically. With the clustered index the data is physically ordered. On any table, there can be as many composite, unique, and non-clustered indexes as needed, but there can be only one clustered index, since this index physically orders the data [MS-SQL93]. It also should be noted that, when multiple indexes are created, the clustered index should be created first.

Defaults, rules, and views are also part of the useful features of the relational model. Default values can be placed in the data holders when no value is specified. When a certain data is entered to be placed in the table, the rules check for the validity of that data for that column in terms of data types. Views help in presenting parts of a full table or database [MS-SQL93].

2.4.1 Entity-Relationship Diagram for the Tool

All elements required for reuse are stored using the relational model. Several interconnected tables are used in this model. Appendix C contains a diagram that represents the entity-relationship diagram for this tool. All the objects that makeup this database are described in the section 3.2

CHAPTER III

DESIGN AND IMPLEMENTATION ISSUES

3.1 Implementation Platform and Environment

The tool was implemented on an IBM compatible personal computer running MS-DOS version 6.0 operating system with the MS-Windows graphical environment. The IBM compatible personal computer was chosen because of its widespread availability, and MS-DOS is arguably the standard operating system for personal computers. MS-Windows environment was chosen to develop the Graphical User Interface (GUI) for this application.

3.1.1 Personal Computers Today

Presently there are two major standards for personal computers - the IBM compatible personal computers and the Macintosh line of personal computers. Recently, in years 1993 and 1994, personal computers based on other CPUs have also been introduced. These computers include PowerPCs and PowerMacs based on the RISC processor jointly developed by IBM, Apple, and Motorola. DEC also introduced a new range of personal computers based on its alpha chip.

The term PC normally refers to the IBM compatible personal computers. The first PCs were developed by IBM with the Intel 8086 processor. This processor has gone through several generations from 8086 to 80286 to 80386 to 80486. The latest generation of this processor is called Pentium (or 80586) which was introduced by Intel in 1993. Presently, the standards for the PC are jointly set by Intel, Microsoft, and several leading vendors of PCs. Since the introduction of the PC in 1981, its power has increased tremendously. Original PCs could handle a maximum of one megabyte of memory, but

they were shipped with far less. Today eight to sixteen megabytes of memory is normally available on most of the new PC's. There has been an attendant increase in the size of the hard disk, and the clarity and the graphics capability of the display also.

3.1.2 DOS Operating System

IBM contracted the Microsoft Corporation to write an operating system for its introductory personal computer, simply called IBM PC. It is estimated that presently DOS runs on one hundred million personal computers world wide. DOS is available in several flavors. The operating system published by Microsoft is called MS-DOS, the one published by Novell (originally by Digital Research) is called DR-DOS, and the IBM product for the operating system is called PC-DOS. Currently the MS-DOS is at Version 6.2.

DOS is a text based operating system with limited support for graphics. The interface of the DOS itself is character based. Several utilities and programs provide the graphics and multi-media capability to the DOS operating system. Version 6.x will be the last version to have the character based interface. One version of DOS from Microsoft has the widely-accepted user interface of Microsoft Windows.

3.1.3 Windows Graphical Environment

Microsoft Windows is a graphical operating environment designed to run on the DOS operating system. Microsoft did not have much success with the graphical interfaces provided by the Windows Versions 1.x and 2.x. One of the chief reasons for this failure was the lack of powerful hardware at the time of their release. But when Microsoft Windows Version 3.0 was released in 1990, its user interface was an instant success. Version 3.1 released in 1992 provided several other capabilities such as multi-media support, object linking, and embedding.

In 1993, Microsoft released a high-end operating system called Windows NT. This is not only a regular operating system (unlike Windows, which requires DOS as the

underlying operating system), but also has several other capabilities such as networking and security. Windows NT, however, retains the user interface of the Windows operating environment.

3.2 Specification and Design of the Tool

The tool that was developed as part of this thesis is named as Reuse with Facets. This tool can be divided into two major parts: the User Interface and the Database.

3.2.1 User Interface

The user interface of the Reuse with Facets tool consists of all the dialog boxes that take the information from the user and present the results. Detailed explanation about the individual dialog boxes and the functionality associated with them is provided in Appendix E titled User and System Administrator Guide. The following paragraphs summarize the user interface provided with the tool.

When this tool, Reuse with Facets, is started by double-clicking the icon provided in the Program Manager, it comes up with a main window titled Reuse with Facets. This window has several menus near the top border, with each menu containing one or more commands. The menus provided here are User, Librarian, Reports, and Help. The User menu has two commands, Retrieval and Insertion. The Librarian menu has five commands named Facet, Vocabulary, Artifact, User, and Role. The Reports menu has two commands, User Report and Artifact Report. Finally, the Help menu contains a command to invoke an "about" box that gives the version-related information about the tool, this also provides the user's general system related information.

Each of these menus evokes a form or a dialog box. Each dialog box has text boxes, list boxes, option buttons, command buttons, and text or graphical instructions to facilitate input from users. Some dialog boxes have tables or grids to present the information to users. The user interface used in this tool conforms to the Microsoft

Windows 3.1 standards and is consistent across the dialog boxes. As mentioned above, all these dialog boxes are thoroughly explained in the user's guide provided in Appendix E.

3.2.2 Database

The database part of the Reuse with Facets tool is a relational database. All the elements required for reuse are stored using the relational model. Several interconnected tables are used in this model. The diagram in Appendix C (titled Relational Schema) represents the entity-relationship diagram for this tool. The following discussion further elaborates the tables used which are tables for Users, Role, Action, Event, Search Results, Facet, Vocabulary, Thesaurus, Artifact.

All the information related to the users is stored in the Users table. Each entry in this table can be uniquely identified with the help of UserId. The RoleId field in this table identifies the title or role of the user in this system. The different roles a user can take are discussed under the Role table. Further, this table stores the First Name, Last Name, Address, and the Telephone Number of users. A comments field is also provided to facilitate additional comments on the user-related information.

The Role Table facilitates the storage of different titles or roles a user can take. Initially, three default roles are provided. Additional roles can be added as the requirement arises. The default roles are: Developer, Manager, and Librarian. The Developer produces and reuses the software artifacts. In general, programmers assume both the roles of developing and reusing. If a situation exists where these two roles are separated, an additional role can be created in the system. The role of Manager is suggested for the person who oversees the complete development process. Managers generally will not produce or reuse the software artifacts, rather they will be responsible for the development at a higher level. And finally, the Librarian is responsible for the maintenance of this system. This role can also be termed as the Super User. The Librarian will constantly update and refine the vocabulary and monitor reuse in the system. The

tool will provide a default user with the role of Librarian, who will work on the further expansion of the system. Hence, this role cannot be deleted from the system.

The Action Table is used to store the names and the descriptions of the actions that a particular user can perform. Three default actions are provided with the tool. The tool provides facilities to add or delete actions. The three default actions that are possible for the reuse events are inserting new software artifacts, retrieving the software artifacts for reuse, and general browsing. The correlation between these actions and the roles mentioned in the above paragraph for a typical situation are: Librarian classifies and inserts the new software artifacts, Developer/Reuser retrieves the existing software artifacts from the system, Manager does the general browsing to get an idea of the existing collection of software artifacts and the amount of reuse.

The Event Table provides a place holder for the information about individual reuse events occurring in the system. Each event can be uniquely identified with the help of an EventId. All the elements that make up an event, i.e., the user of the system, the role of this user, the action this user performed, the string of facet ids and the vocabulary associated with each facet, the result this search operation has ended up with, the artifact involved in the search, the date and time of the event, and any comments about the event, are stored in this table. All these events can be summarized later in the form of reports. These reports provide a number of software performance and usage metrics about the system.

The SearchResult table contains all the possible outcomes of the search for an artifact. These outcomes are Found, Not Found, or Aborted. Aborted is reported for a search result when the user closes the system or the query while the search is still being performed.

The Facet and Vocabulary tables make up the necessary words required for the classification of software artifacts. The tool provides several default facets and vocabulary for the PC applications domain. The Facet table is the holder of the name of

the facet and a description associated with it. The Vocabulary Table provides the storage for the different words that describe each facet. Since a vocabulary word is associated with a particular facet, a FacetId field is also added to this field.

The Thesaurus table can be used to obtain synonyms for the vocabulary used for the artifacts in the system. Even though this table provides synonyms for the vocabulary mentioned in the previous table, these two tables are separate. The reason for this separation is that a word or synonym still remains in the system even after a particular facet or vocabulary is removed from the system. This ensures the growth of the thesaurus.

The Artifact table stores the classification and the physical details of the artifact. The FacetId string and the VocabularyId string give the classification. ArtifactName, ArtifactFileName, ArtifactPath, ArtifactSize, ArtifactDescription, ArtifactAuthor, and ArtifactDateOfInsertion provide the physical details of the system.

3.3 Development and Implementation of the Tool

The graphical interface for this application is developed in Microsoft Visual Basic Version 3.0. The relational database which holds the information and classification about the reusable software artifacts is developed using on Microsoft Access Version 1.1 database engine. For this database development, more sophisticated database environments such as the MS-SQL Server were considered. Since the SQL Server does not provide executables (i.e., the SQL Server should be present on the computer on which an application is running), Microsoft Access was given priority as the database engine. Also, Microsoft Access 1.1 is fully integrated into Microsoft Visual Basic 3.0 for all its database features.

A setup program was also developed to ease the setting up of this tool on a reuser's workstation. This tool installs a default database which can be moved to a network directory after the setup. Step-by-step instructions are provided in the user's

guide presented under Appendix E. This tool was tested for insertion and retrieval of software artifacts. It was also tested for all the system administration functions (or the Librarian functions) concerning updating the list of facets, vocabulary, artifacts, thesaurus, users, and roles.

Code for this program is broken in to several forms and module files. Each form contains the instance code for all the user interface elements like command buttons, menus, and the form appearance itself. As this is event driven programming, each user interface element contains the code module that is to be executed when certain event occurs. Examples for an event are mouse click, mouse double click, pressing a key, etc. All the form functions are local to that form, where as module files contain global functions.

All the functions in this program contain a header. Comments are intermixed with the code to make it readable.

CHAPTER IV

TESTING OF THE TOOL

4.1 Sample Reuse Done with the Tool

This chapter briefly outlines a sample deposit and retrieval. A number of observations are also mentioned. Figure 4-1 depicts a function that changes the string of a color name to a hexadecimal color code. This function can be used to obtain a color name from the user and then generate the color code to actually use that color. So, this is mainly a user interface related function. And further, this is a color related function.

This function is written in Visual Basic and runs under the Windows operating system. The Name of this function is GetColorCode and is written by Mareddy on 01/01/94. So the class for this function according to the Faceted Classification System is Windows-VB-UserInterface-Color-GetColorCode-Mareddy-010194. This name is obtained by the synthesis of all the facets for the function.

For the retrieval of this component, the same class (with most of the facets) is inputted into the system. For example, if the Action facet for this artifact is selected as "Shade" instead of "Color", the system will miss this artifact. But in the system Shade is also defined as a synonym for color. So, a further prompt will be given to the reuser, asking whether to look for classes for synonymous facets. Subsequently, the tool can retrieve the above artifact.

```
Function GetColorCode (sColorName As String) As Long
    '-------------
    'Declarations
    '-------------
    Dim sMessage, sTitle As String
    CONST G_BLACK=0, G_BLUE=1, G_GREEN=2, G_CYAN=3
    CONST G_RED=4, G_MAGENTA=5, G_BROWN=6, G_LIGHT_GRAY=7
    CONST G_DARK_GRAY=8, G_LIGHT_BLUE=9, G_LIGHT_GREEN=10, G_LIGHT_CYAN=11
    CONST G_LIGHT_RED=12, G_LIGHT_MAGENTA=13, G_YELLOW=14, G_WHITE=15
'---------------------------------------
'Get the color code for color string
'---------------------------------------
Select Case (sColorName)
    Case ("BLACK")
        GetColorCode = QBColor(G_BLACK)
    Case ("BLUE")
        GetColorCode = QBColor(G_BLUE)
    Case ("GREEN")
        GetColorCode = QBColor(G_GREEN)
    Case ("CYAN")
        GetColorCode = QBColor(G_CYAN)
    Case ("RED")
        GetColorCode = QBColor(G_RED)
    Case ("MAGENTA")
        GetColorCode = QBColor(G_MAGENTA)
    Case ("BROWN")
        GetColorCode = QBColor(G_BROWN)
    Case ("LIGHT GRAY")
        GetColorCode = QBColor(G_LIGHT_GRAY)
    Case ("DARK GRAY")
        GetColorCode = QBColor(G_DARK_GRAY)
    Case ("LIGHT BLUE")
        GetColorCode = QBColor(G_LIGHT_BLUE)
    Case ("LIGHT GREEN")
        GetColorCode = QBColor(G_LIGHT_GREEN)
    Case ("LIGHT CYAN")
        GetColorCode = QBColor(G_LIGHT_CYAN)
    Case ("LIGHT RED")
        GetColorCode = QBColor(G_LIGHT_RED)
    Case ("LIGHT MAGENTA")
        GetColorCode = QBColor(G_LIGHT_MAGENTA)
    Case ("YELLOW")
        GetColorCode = QBColor(G_YELLOW)
    Case ("WHITE")
        GetColorCode = QBColor(G_WHITE)
    Case Else
        'Error, ini file has wrong color name.
        sMessage = "Wrong Color code."
        sTitle = "InOut Board - GetColorName"
        MsgBox sMessage, MB_OK, sTitle
        GetColorCode = G_WRONG_COLOR
End Select
End Function
```

Figure 4-1. GetColorCode Function

4.2 Observations

The following observations were made during the testing of this tool, Reuse with Facets.

- Initially, the number of hits for a sample class were few. The reason for this was that the number of artifacts present in the system was small. As the repository of software artifacts grew, the chance of retrieving an artifact with desired classification increased.

- This author depended heavily on memory for reusing the existing artifact, in terms of this locations and names. But now, a convenient system, i.e., the Reuse with Facets tool, existed with a classification system and a friendly user interface.

- Reports provide a way to measure the amount of reuse happening in an organization. The functions that are not being used over a period of time can be deleted and popular artifacts can be left in the system for reuse.

- A new specification called OLE (Object Linking and Embedding) has some similarity with this tool. Microsoft Windows comes with a large set of functions called Windows API (Application Programming Interface). These functions let the programmer access the operating system related functionality. These functions are randomly named and grouped, making Windows API chaotic and hard to use. A new specification called OLE for personal computer operating systems defines the communication between objects. Objects adhering to this standard are called Windows Objects and will be able to communicate effectively. Each object will be able to access the functions of other objects without knowing the implementation details.

Some limitations were also observed regarding this tool.

- This tool works only in the Microsoft Windows environment. This tool is not readily portable to other operating systems.

- This tool is passive in identifying new artifacts, extracting new facets, and verification of the inputs.

- Conceptual distance between two facets is enforced through synonyms, rather than by assigning weights to terms.

CHAPTER V

SUMMARY, CONCLUSIONS, AND FUTURE WORK

This thesis and the tool described mainly dealt with the implementation of the Faceted Classification System for reuse in the Personal Computer applications development domain. As part of this effort, a database schema, a list of facets, and a vocabulary pertinent to this domain were developed. The database developed conforms to the standard relational model.

This model of software reuse can be effectively used in a multitude of development situations. Pure software development corporations, with development projects in several platforms, can benefit the most from such a tool. But so will the Information Systems divisions of the corporations where software development is not the main business. They will be able to save time and resources with the help of an effective classification system and a reuse tool.

The success of reuse depends on how strong the organizational commitment is to reuse. In the case of the Faceted Classification System, it is equally important to develop and maintain a pertinent list of facets and vocabulary.

Future work to extend this thesis can be towards addressing the limitations mentioned at the end of Chapter III. A domain-independent tool can help the multi-platform development tremendously, thus increasing reuse across domains. More intelligence can also be built into the tool, thus facilitating the identification of artifacts and verification of inputs. Adding capabilities such as operators and composition rules to facilitate the fabrication of new software artifacts based on or by combining existing artifacts.

REFERENCES

[Barnes87] B. Barnes, T. Durek, J. Gaffney, and A. Pyster, "A Framework and Economic Foundation for Software Reuse", *Proceedings of the RMISE Workshop on Software Reuse*, Rocky Mountain Institute of Software Engineering, Boulder, Colorado, pp. 77-88, October 1987.

[Biggerstaff87] T. Biggerstaff and C. Richter, "Reusability: Framework, Assessment, and Directions", *IEEE Software*, pp. 41-49, July 1987.

[Burton87] B.A. Burton, R.W. Aragon, S.A. Bailey, K.D. Koehler, and L.A. Mayes, "The Reusable Software Library", *IEEE Software*, pp. 25-33, July 1987.

[Date91] C.J. Date and H. Darwen, *Relational Database Writings 1989-1991*, Addison-Wesley Publishing Company, Massachusetts, 1991.

[Hall87] P.A.V. Hall, "Software Components and Reuse - Getting More out of Your Code", *The International Journal of Information and Software Technology*, Vol. 29, No. 1, pp. 38-43, January/February 1987.

[Hall92] P.A.V. Hall, *Software Reuse and Reverse Engineering in Practice*, Chapman and Hall, New York, 1992.

[Immroth71] J.P. Immroth, *A Guide to the Library of Congress Classification*, Libraries Unlimited, Colorado, 1971.

[Katz87] S. Katz, C.A. Richter, and K.S. The, "PARIS: A System for Reusing Partially Interpreted Schemas", *Proceedings of the Ninth Annual International Conference on Software Engineering* , Washington, D.C., pp. 377-385, March/April 1987.

[Krueger92] C. Krueger, "Software Reuse", *ACM Computing Surveys*, Vol. 24, No. 2, pp. 131-183, June 1992.

[Lenz87] M. Lenz, H.A. Schmid, and P.W. Wolf, "Software Reuse through Building Blocks", *IEEE Software*, pp. 34-42, July 1987.

[MS-SQL93] *Microsoft SQL Server Implementation Notes*, Microsoft University, 1993.

[Prieto-Diaz91] R. Prieto-Diaz, "Implementing Faceted Classification for Software Reuse", *Communications of the ACM*, Vol. 34, No. 5, pp. 89-97, May 1991.

[Swanson92] J. E. Swanson and Mansur H. Samadzadeh, "A Reusable Software Catalog Interface", *Proceedings of the 1992 ACM/SIGAPP Symposium on Applied Computing (SAC'92)*, pp. 1076-1082, Kansas City, MO, March 1992.

[Tracz87] W. Tracz, "Ada Reusability Efforts: A Survey of the State of the Practice", *Proceedings of the Fifth Annual Joint Conference on Ada Technology and Washington Ada Symposium*, U.S. Army Communications-Electronics Command, Ft. Monmouth, N.J. pp. 35-44, March 1987.

[Tracz88a] W. Tracz, "Software Reuse Myths", *ACM SIGSOFT Software Engineering Notes*, Vol. 13, No. 1, pp. 17-21, January 1988.

[Tracz88b] W. Tracz, "RMISE Workshop on Software Reuse: Meeting Summary", *Tutorial on Software Reuse: Emerging Technology*, Boulder, CO, pp. 41-53, October 1988.

[Transact-SQL93] *Transact-SQL User's Guide for SQL Server*, Microsoft Corporation, 1993.

[Zand92] M. K. Zand, Mansur H. Samadzadeh, H. Saiedian, and H. Farat, "Classification and Identification of Software Components", *Proceedings of the Second Golden West International Conference on Intelligent Systems*, pp. 275-280, Reno, NV, June 1992.

[Zand93] M. K. Zand, K. M. George, Mansur H. Samadzadeh, and H. Saiedian, "An Interconnection Language for Reuse at the Template/Module Level", *The Journal of Systems and Software*, Vol. 23, No. 1, pp. 9-26, October 1993.

[Zand93] M. K. Zand, Mansur H. Samadzadeh, and H. Saiedian, "Version Management for ROPCO: A Micro-Incremental Reuse Environment", *The Journal of Information and Software Technology*, Vol. 35, No. 11/12, pp. 627-637, November/December 1993.

APPENDIX A

GLOSSARY

Ada	A programming language developed under the auspices of the US Department of Defense to alleviate the "Software Crisis".
Client-Server Model	In this model, a central computer holds the databases or files and there are several workstations connected to this computer that have access to those databases and files. The central computer is called the server and all the workstations are called clients.
Column	Each column or field describes one characteristic of the entity (e.g., a person's name or address, a company's name or president, items sold, a quantity, or a date).
Database Engine	A program that facilitates the definition, development, and querying of a database. A typical database program contains two parts: a front end for the user interface and a back end for holding the database.
Dewey Decimal System	A classification system, mainly used in US libraries for organizing books. In this system, all classes are predefined.
Domain	An area or platform of development. For example, UNIX and DOS are two major platforms or domains for software development on mainframes and personal computers, respectively.
Facet	An attribute of a domain. For example, UNIX tools may be described by these four facets: by action, by object, by data structure, and by system.
Faceted Classification System	A classification system that is widely used in the libraries of India and Europe. Here, classes are

	obtained from various faceted lists for a particular domain.
Foreign Key	A column that provides a link with another table. These two columns should be of identical data types.
Graphical User Interface (GUI)	A non-character based interface to a programming application. Commands to the program are typically given through menus, buttons, dialog boxes, and other graphical elements.
Librarian	A person organizing and maintaining the software artifacts that will be used in future software projects.
Pipes and Filters	Inputting one program's output to another program, where two programs are on the same command line with a pipe (|) in between. Filter commands divide, rearrange, or extract portions of the information that passes through them (for e.g., more, find, and sort)
Primary Key	The column or group of columns whose values uniquely identify each row in the table.
Relational Model	A data organization and retrieval model first proposed by Dr. E. F. Codd in 1960's. This model uses tables and only tables at the logical level to organize data.
Reuser	A programmer building a software system who uses software artifacts from a previously developed system.
Row	Each row or record of a table describes one occurrence of an entity (e.g., a person, a company, or a sale).
SQL	Structured Query Language, a language originally developed at IBM in 1970's based on the relational model.
Table	Tables are logical data structures in the relational system. Each table contains rows and columns.

APPENDIX B

TRADEMARK INFORMATION

IBM	IBM is the registered trademark of International Business Machines.
MS-DOS	MS-DOS is the registered trademark of Microsoft Corporation.
MS-Windows	MS-Windows is the registered trademark of Microsoft Corporation.
MS-Visual Basic	MS-Visual Basic is the registered trademark of Microsoft Corporation.
MS-Access	MS-Access is the registered trademark of Microsoft Corporation.
MS-SQL Server	MS-SQL Server is the registered trademark of Microsoft Corporation.
DEC	DEC is a registered trademark of Digital Equipment Corporation.
MacIntosh	MacIntosh is a registered trademark of Apple Computer Corporation.
Motorola	Motorola is a registered trademark of Motorola Corporation.

APPENDIX C

RELATIONAL SCHEMA

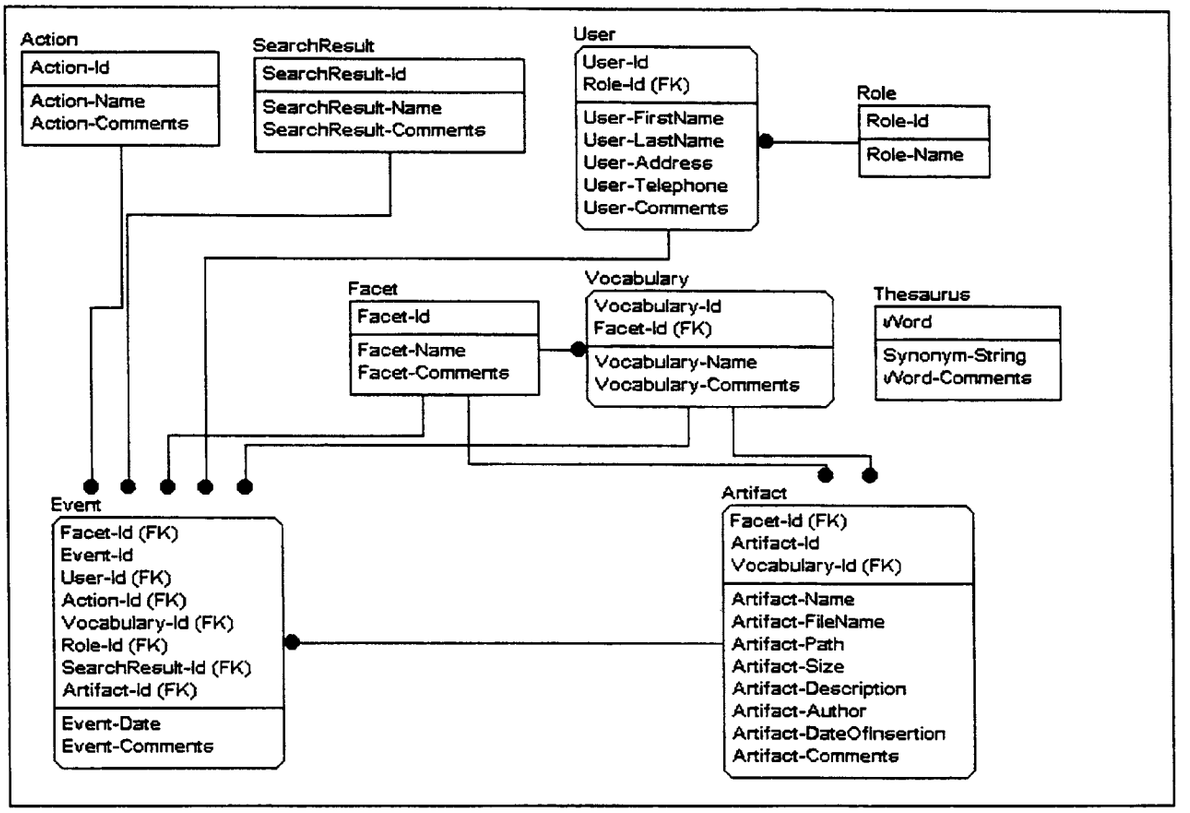

Figure C-1. Entity-Relationship Diagram for the Reuse with Facets tool

APPENDIX D

SAMPLE DATABASE

Number	Name of the Facet
1	Operating System
2	Language
3	System Component
4	Action
5	Name
6	Author
7	Date

Figure D-1. Sample List of Facets

Number	Name of the Role
1	Developer/Reuser
2	Librarian
3	Manager

Figure D-2. Sample List of Roles

Operating System	Language	System Component	Action	Name	Author	Date
DOS	C	Specificatio	Average			
Windows	C++	n	Count			
Windows	Pascal	Data Model	Color			
NT	FORTRAN	User	Shade			
OS/2	VB	Interface	Max			
UNIX	QB	Editor	Min			
None	COBOL	Financial	Sum			
All	Executable	Database	Len			
	None	Initialization	Lower			
	All	Formatting	Upper			
		Date &	Cos			
		Time	Sin			
		Math & Trig	Tan			
		Statistical	Create			
		Text	Insert			
		Logical	Load			
		Engineering	None			
		None	All			
		All				

Figure D-3. Sample Facet and Vocabulary List

APPENDIX E

USER AND SYSTEM ADMINISTRATOR GUIDE

1. Introduction

Welcome to Reuse with Facets, an effective tool to promote the reuse of software artifacts in your organization. This tool uses the faceted classification system and an intuitive user interface to facilitate the organization and retrieval of software artifacts created by your programmers and software development personnel. The following sections describe the setup and various features of the tool. This guide is prepared to address the needs of both the Librarian and the Developer. Since the functions of the Developer (as compared to the functions of the Librarian) do not include the classification-related tasks and the generation of reports for the purpose of software performance and usage metrics, user may refer to the sections related to the retrieval of software artifacts only.

2. Setting up

The Reuse with Facets tool can be installed on your system by using the program setup.exe. The setup program installs on your hard disk the main interface program, the database with the default facets, the vocabulary, roles, search results, the thesaurus, and other system related variables. Since this tool is written for Microsoft Windows, it installs several system related files on your hard disk. Also, since this tool is developed for reuse in PC applications development, all these defaults reflect the PC development domain.

2.1 Hardware and Software Requirements

Before you install Reuse with Facets, make sure your computer meets the following minimum requirements in terms of hardware and software.

- Any IBM-compatible machine with an 80386 or higher processor.
- A hard disk with at least 2MB of free space. More disk space will be required as the collection of software artifacts grows.
- A 5 1/4 or 3 1/2 inch floppy disk drive.
- A VGA monitor or better.
- Two megabytes of memory (four megabytes recommended for speed).
- A network. Even though this tool works for a single-user setup, when multiple people are involved in the software development process, a network and a central server to store the software artifacts will be required.
- A mouse.
- MS-DOS or PC-DOS version 3.1 or later.
- Windows version 3.1 in enhanced mode.

2.2 Running the Setup

Follow the steps belong to install the Reuse with Facets tool on your workstation. Do not copy the files directly onto your system, since they are in the compressed mode.

1. Insert Disk #1 in drive A.
2. From the File menu of the Program Manager or File Manager, choose Run.
3. Type a:setup
4. Follow the Setup instructions on the screen.

During setup, you will be asked to provide the path of the directory where you want to install the files related to the tool. This setup program will install all the files, including the database files, in this directory. If multiple users are going to use the tool, move the database-related files, i.e., all files with extensions mdb and ldb, to a central location. Then change the default data source entry in the initialization file (rwf.ini) to reflect this change.

When multiple users are going to use the tool, the setup program must be run on each of these workstations. Since all these users are going to use the database at the central location, the database files created on the users' workstations must be deleted. You may also use several different databases (for different domains) and change the initialization file entry to correspond to this.

3. Insertion of Software Artifacts into the System

When the tool is started, the user is forced to login. This is, however, used not to enforce security but to obtain the name of the user. This information is used later in compiling the usage reports. A picture of the Login box is provided in Figure E-1.

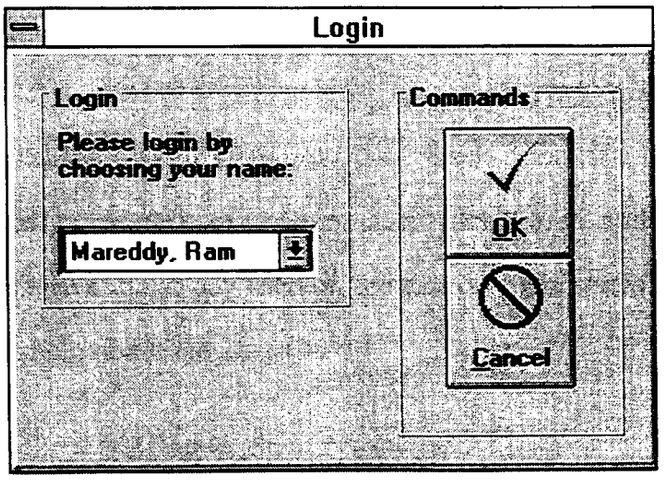

Figure E-1. Login box

3.1 Main Window

All the functionality of the Reuse with Facets can be accessed through the main window with the help of menus and tool buttons provided there. Figure E-2 is a representation of this window.

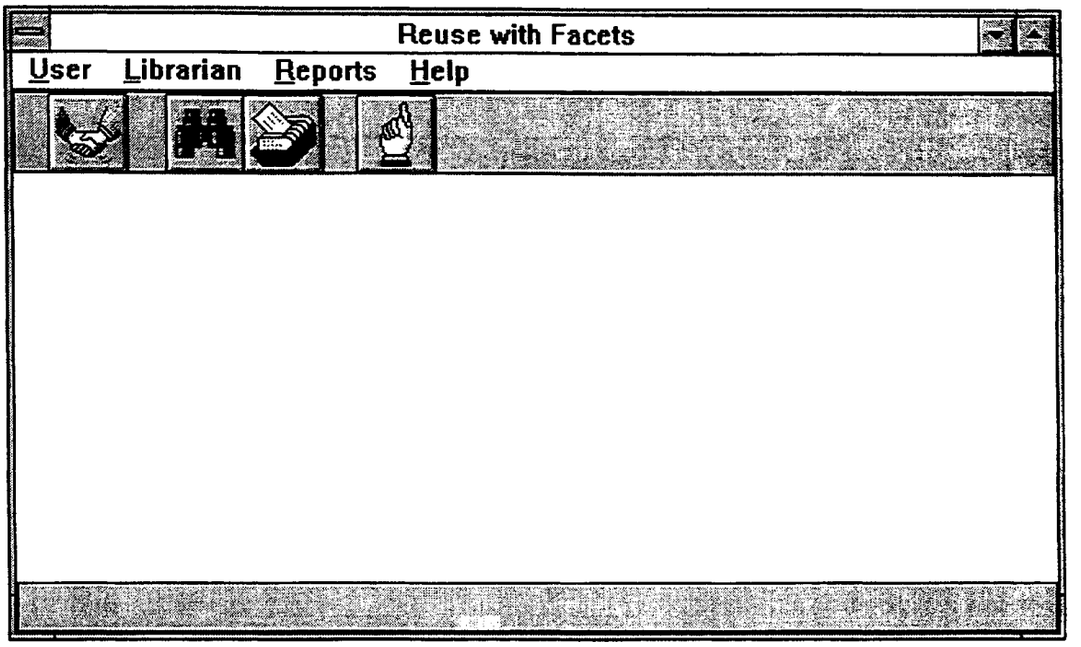

Figure E-2. Main Window of Reuse with Facets

The menus attached to this window, i.e., User, Librarian, Reports, and Help, provide the functionality of this tool. Commands like search and insert can also be accessed through the tool buttons provided on the tool bar. The status bar at the bottom of the window provides help about the items.

3.2 Preparation of Software Artifact

When a Developer or the Librarian comes across a piece of software artifact that is potentially reusable in a later software development, the Developer/Librarian should

first prepare the artifact to insert into the system. A software artifact need not necessarily be a code fragment, it can be a specification, a documentation, a design structure, a module-level implementation structure, or any other software component. This artifact must be saved (after any appropriate separation or modification) in a file. Then choose the Insert command from the User menu of the tool. The dialog box shown in Figure E-3 will be presented to the user.

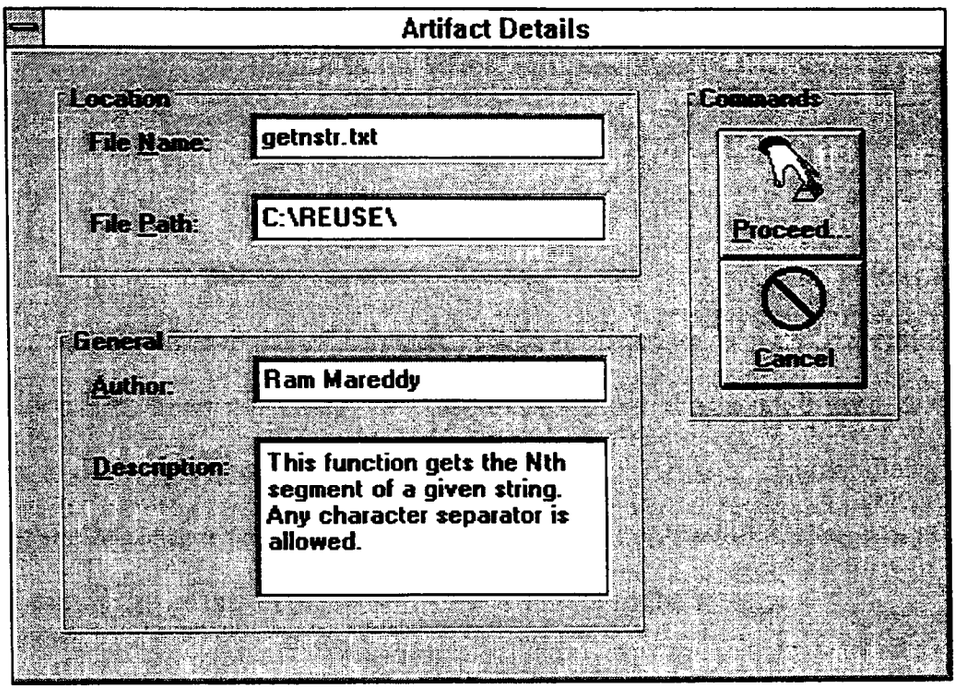

Figure E-3. Artifact Details

In this dialog box (Figure E-3) the location and other general details are requested from the user. Location details, i.e., the name of the file in which the software artifact under consideration is saved and the path to reach that file, must be inputted by the user. The tool will check these details to make sure that the file exists at the aforementioned

place. Entry of the general details, i.e., the author of the artifact and a description, is not mandatory, but highly recommended.

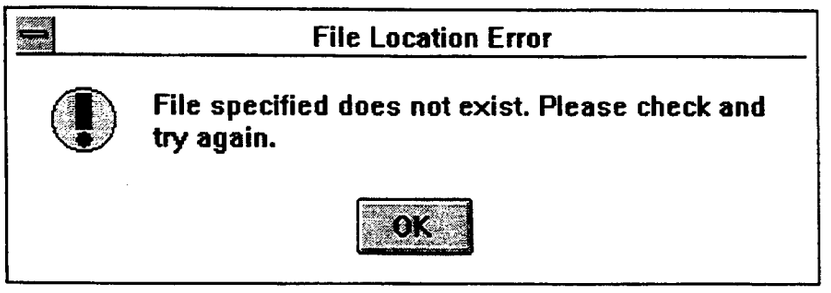

Figure E-4. Artifact Existence Error

After these details are entered, the user needs to click the button captioned Proceed..., which will take the user to the classification screen. If the file mentioned in the above dialog box does not exist, an error message (as shown in Figure E-4) will be displayed and the processing will come back to the Artifact Details dialog box for correcting the name and path of the file. The cancel button on this dialog box provides an option to cancel the insertion of the software artifact into the system.

3.3 Classification of Software Artifact

If the software artifact mentioned in the previous dialog box does exist in the system, the next step is to classify that artifact according to the facets and the associated vocabulary. For this purpose, a dialog box similar to Figure E -5 will be presented to the user.

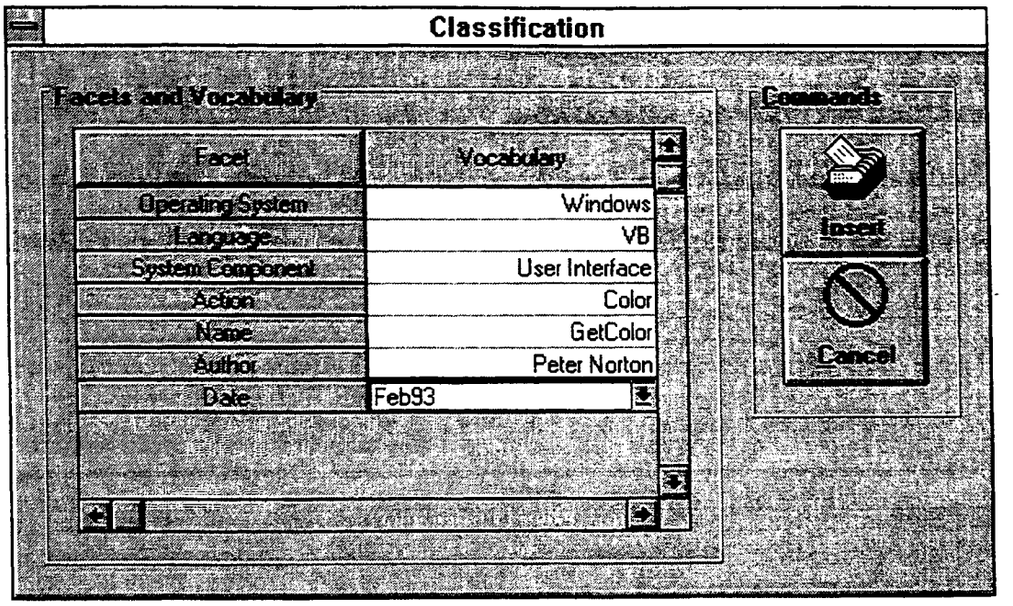

Figure E-5. Classification of the Artifacts

The first column in Figure E-5 contains the list of facets with which this software artifact can be described. For each of these facets there is another list of vocabulary in the second column. Click on the pull-down button on these list boxes to choose the appropriate word to describe that particular facet. Once the selection of vocabulary is completed, click on the Insert button to insert this artifact into the system with the classification described on the table in Figure E-5.

4. Retrieval of Software Artifacts from the System

This section is of particular interest to Developer/Reuser rather than the Librarian. Hence, this section can be treated as the Reuser's guide and rest of this guide can be taken as the Librarian's Guide. The process of obtaining an artifact from the repository contains two steps, search and retrieval.

4.1 Search of the Repository

Figure E-6 depicts the dialog box presented to the user for the purpose of searching for software artifacts which match a given specification. This functionality can be obtained by choosing the command Retrieval from the User menu in the main window of the tool.

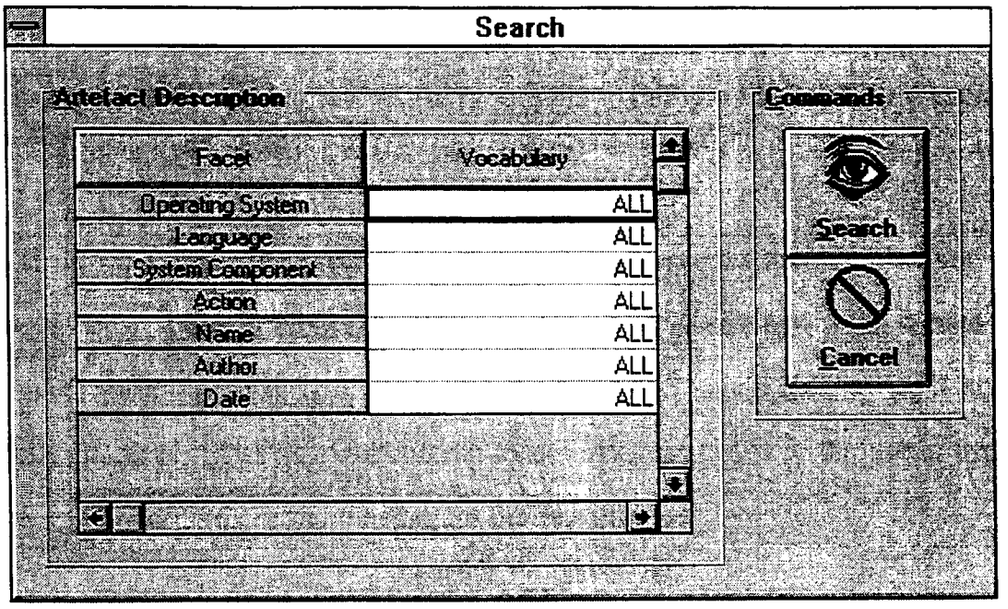

Figure E-6. Search of the Artifacts

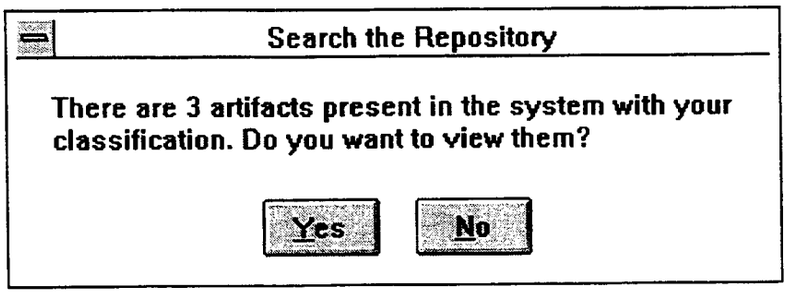

Figure E-7. Hits in the Search of Repository

This dialog box (Figure E-6) is similar to the classification dialog box. The first column contains the list of all facets available to describe an artifact. The second column contains the vocabulary for each artifact. After the selection of vocabulary is completed, click on the Search button to search for the artifact with given class.

The search of the repository can result in a hit of one more artifacts, or there can be a miss, i.e., no artifact available in the system matching the description given by the user. If there are hits as shown in Figure E-7, the tool will present a dialog box similar to the one in Figure E-10 for the retrieval of the artifact, otherwise (if there are no hits, as shown in Figure E-8) a dialog box to use synonyms as in Figure E-9 will be presented.

4.2 Synonyms

If the user chooses to search the system with synonyms, an extra column will be added to the search dialog box. This column provides the flexibility to the user to use the synonyms for an individual facet rather than for all facets. Figure E-9, which is similar to the search dialog box, has an extra column to indicate Yes or No to the use of synonyms for the corresponding facet and vocabulary in the columns.

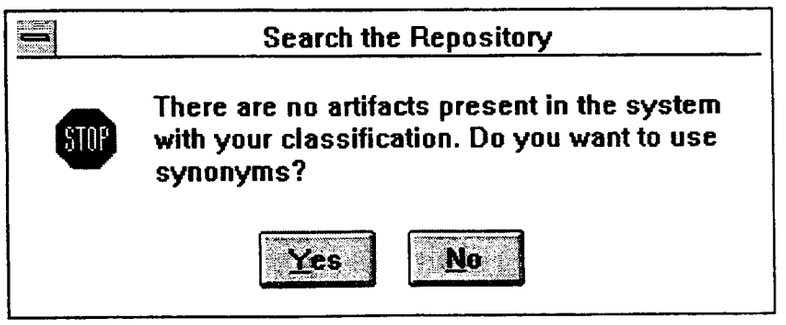

Figure E-8. Suggestion for synonyms

More Yes's in the synonyms column (in Figure E-9) will take more time to search the repository for artifacts. This will also broaden the search criteria and there will be more chance of a hit for an artifact. After the search, if there are any hits in the system, they will be presented in the Retrieval dialog box (Figure E-10) to retrieve the artifact from the system, else the synonyms dialog box will be presented to further broaden the scope of search.

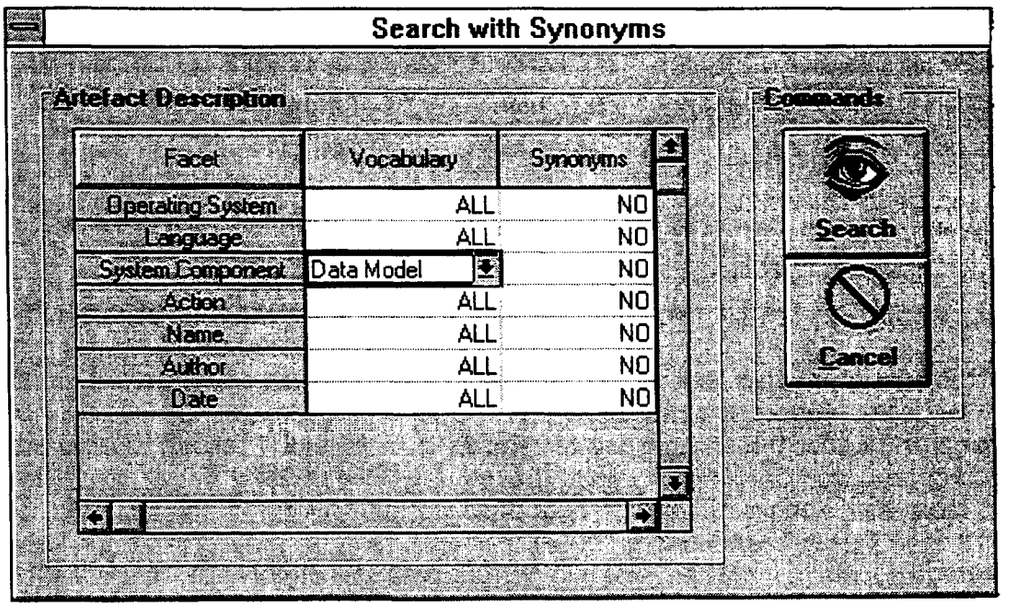

Figure E-9. Search with Synonyms

4.3 Retrieval of the Artifacts

If any hits result from the search for software artifacts, they will be presented in the retrieval dialog box with all the physical characteristics.

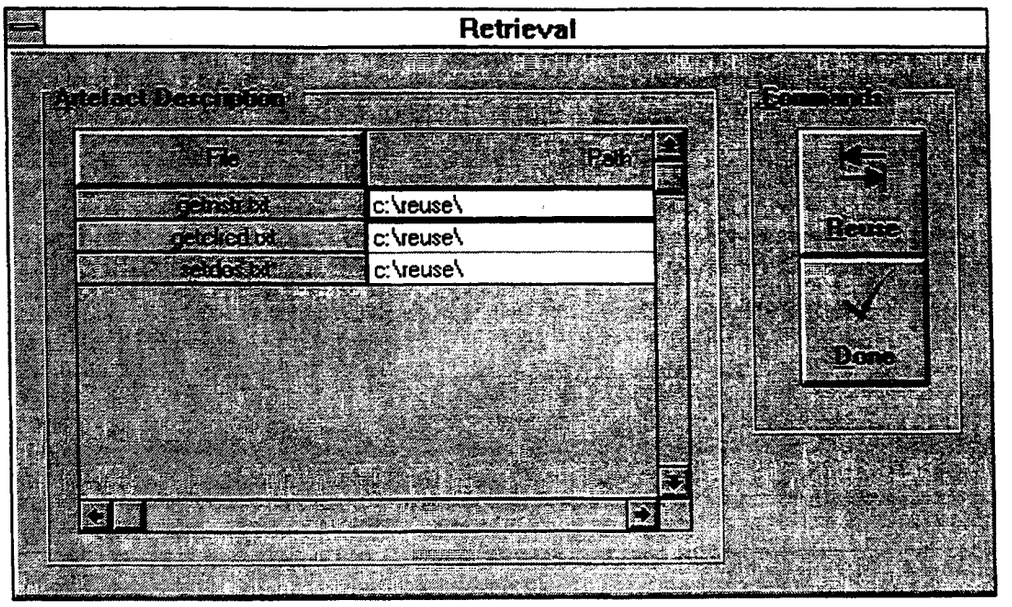

Figure E-10. Retrieval of the Artifacts

The Retrieval dialog box presents the filename, path of the file, author, and date of insertion. By using the details in this screen, the reuser then can choose the artifact that meets the present needs. When the user decides to use a particular artifact, it is requested that user click on the command button captioned Reuse. This click will store it as an event with the currently-logged-in user and the currently-selected artifact as participants. This information will be used in the generation of the performance reports.

5. System Administration

System administration is a Librarian-specific function. All the necessary commands are available from the Librarian menu in the main window. These commands provide the ability to the Librarian to add or delete facets, add or delete a vocabulary item, delete the artifact entry in the database, and add or delete the entries in the thesaurus. A second set of commands is provided to administer the user-related issues. These commands provide the ability to the Librarian to add or delete users, and add or delete roles.

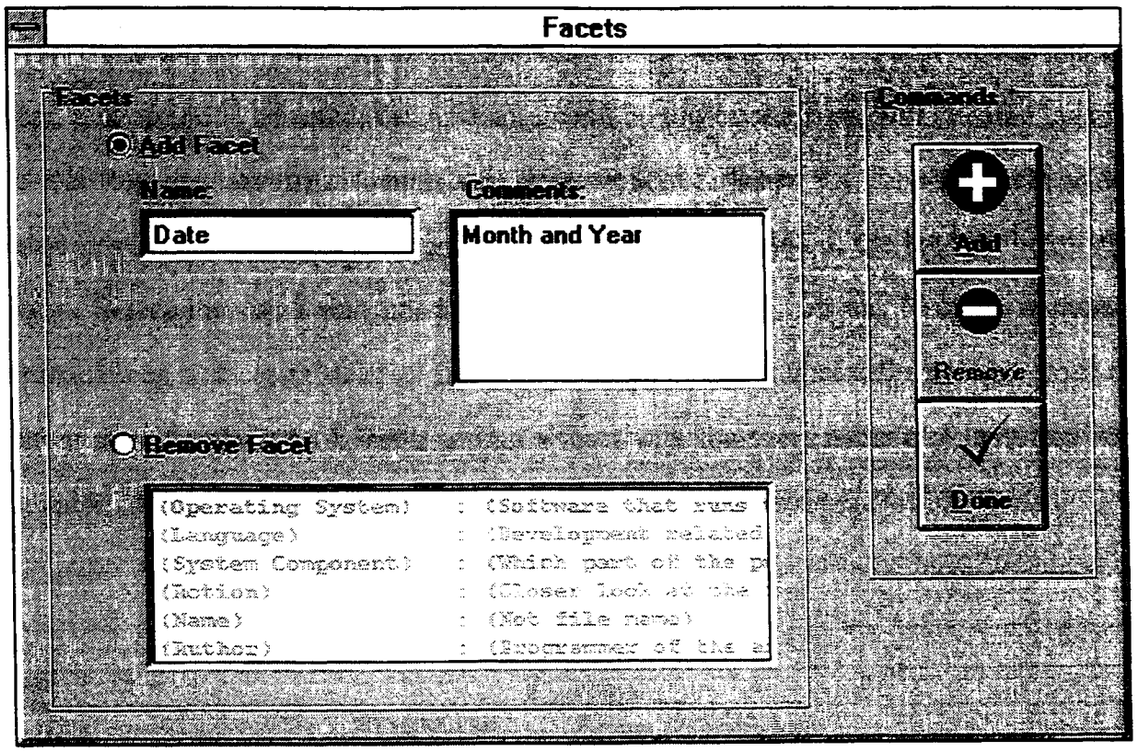

Figure E-11. Administration of Facets

5.1 Facets

To add or remove facets, choose the Facet command from the Librarian menu. A dialog box similar to Figure E-11 will be presented to facilitate the administration of facets.

To add a facet to the system, select the Add Facet option button. When this option is selected, the Add command button on the right will become active and the Remove command button will become inactive, i.e., it will be grayed out. With the Add Facet option button selected, input the name of the facet and a brief description in the two text boxes provided below the option button. Then, to add this facet to the system, click on the Add button.

To remove a facet from the system, choose the Remove Facet option button. When this option is selected, the Add command button on the right will become inactive and the Remove command button will become active. When the Remove facet option button is selected, a list of the existing facets will be presented in the list box below this option. Select the facet that needs to be deleted and click on the Remove command button. When a facet is deleted from the system, it is automatically deleted from the artifact classification. As a result adding a facet with the same name at a later time will not automatically restore the earlier classification for the existing artifacts.

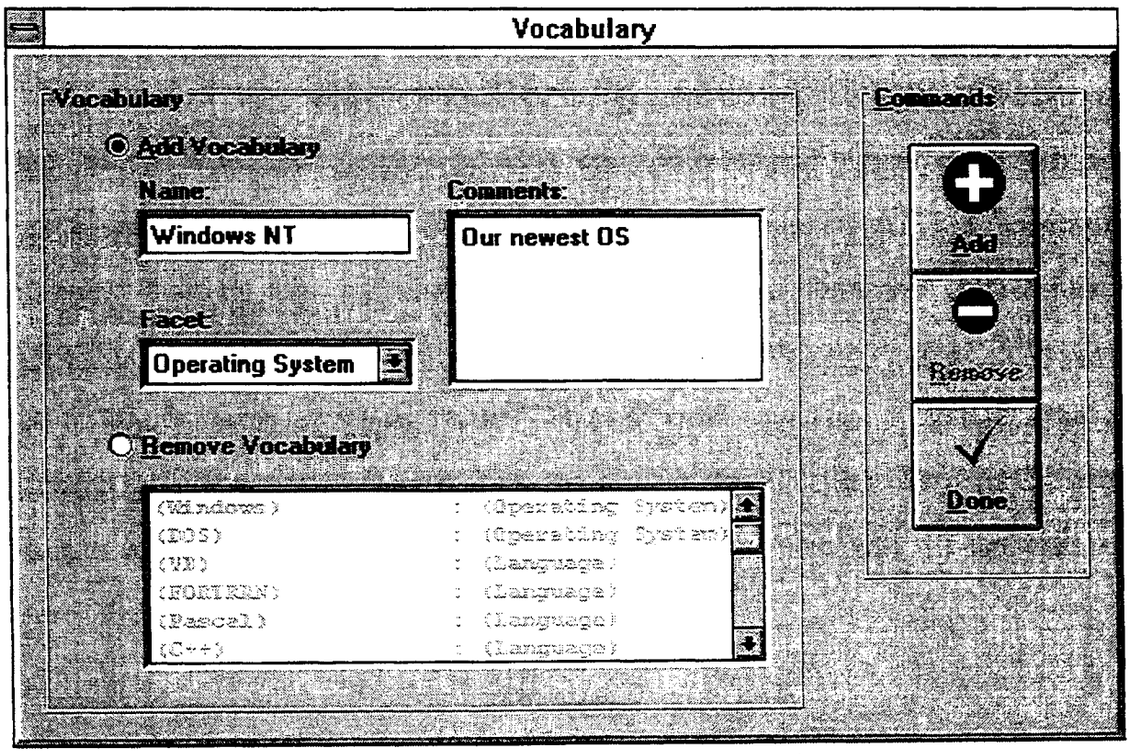

Figure E-12. Administration of Vocabulary

5.2 Vocabulary

To add or remove a vocabulary item, choose the Vocabulary command from the Librarian menu. A dialog box similar to Figure E-12 will be presented to facilitate the administration of the vocabulary.

To add a vocabulary term to the system, select the Add Vocabulary option button. When this option is selected, the Add command button on the right will become active and the Remove command button will become inactive, i.e., it will be grayed out. With the Add Vocabulary command button selected, input the name of the vocabulary term, choose the facet that this term will describe, and a brief description in the two text boxes, and a selection box presented below. Then to add this vocabulary term to the system, click on the Add button.

To remove a vocabulary term from the system, choose the Remove Vocabulary option button. When this option is selected, the Add command button on the right will become inactive and the Remove command button will become active. When the Remove Vocabulary button is selected, a list of the existing vocabulary terms will be presented in the list box below this option. Select the vocabulary term that needs to be deleted and click on the Remove command button. When a vocabulary term is deleted from the system, it is automatically deleted from the artifact classification. As a result, hence adding a vocabulary term with the same name at a later time will not automatically restore the earlier classification for the existing artifacts.

5.3 Artifact

To remove an artifact from the classification, choose the Artifact command form the Librarian menu (to insert an artifact into the system, use the Insert command from the User menu). A dialog box similar to Figure E-13 will be presented to the user to facilitate the administration of the Artifacts.

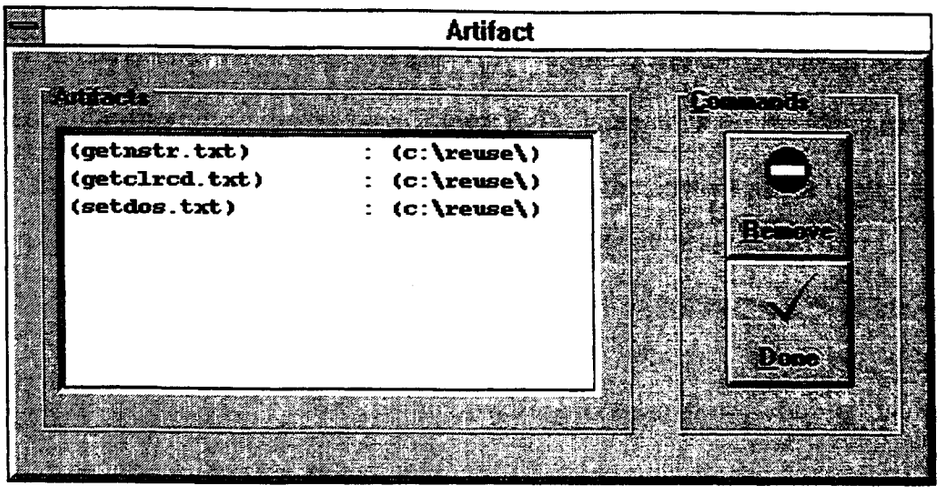

Figure E-13. Administration of Artifacts

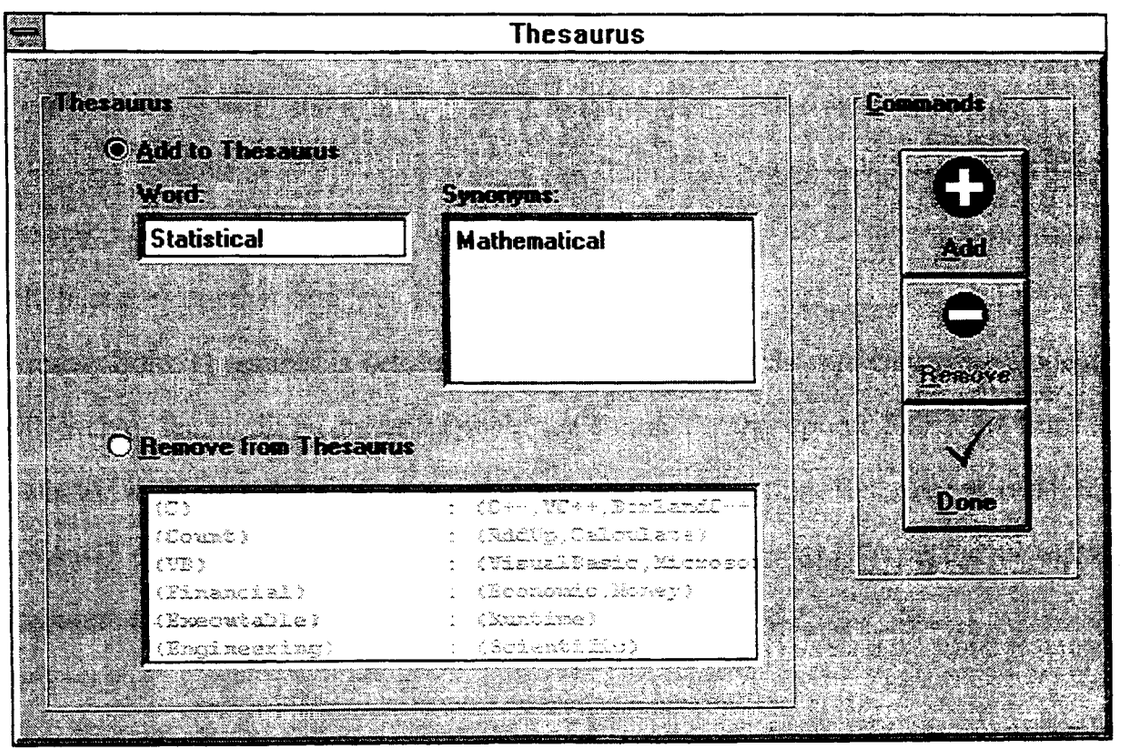

Figure E-14. Administration of Thesaurus

To remove an artifact from the system, first select the artifact from the list box in the Artifact Administration dialog box (Figure E-1). Then click on the Remove command button on the right. This will only remove the entry of the artifact in the tables of the tool, i.e., it will not delete the physical file from the hard disk.

5.4 Thesaurus

To add or remove a word-synonym string combination, choose the Thesaurus command from the Librarian menu. A dialog box similar to the Figure E-14 will be presented to the user to facilitate the administration of thesaurus items.

To add a Word-Synonym String to the system, select the Add to Thesaurus option button. When this option is selected, the Add command button on the right will become active and the Remove command button will become inactive, i.e., it will be grayed out. With the Add to Thesaurus option button selected, input the word and the string of synonyms, separated by commas, in the two text boxes provided below the option button. Then, to add this word and synonym string to the system, click on the Add button.

To remove a Word-Synonym string from the system, choose Remove from Thesaurus option button. When this option is selected, the Add command on the right will become inactive and the Remove command button will become active. When Remove from Thesaurus is selected, a list of existing words and synonyms associated with each of them will be presented in the list box below this option. Select the word that needs to be deleted and click on the Remove command button. Removal or addition of terms will not result in any changes to the classification of the artifacts, as the thesaurus is a separate entity from rest of the database.

5.5 User

To add a user to the system with all the accompanying information, or remove a user from the system, choose the User command from the Librarian menu. A dialog box

similar to Figure E-15 will be presented to the Librarian to facilitate the administration of users.

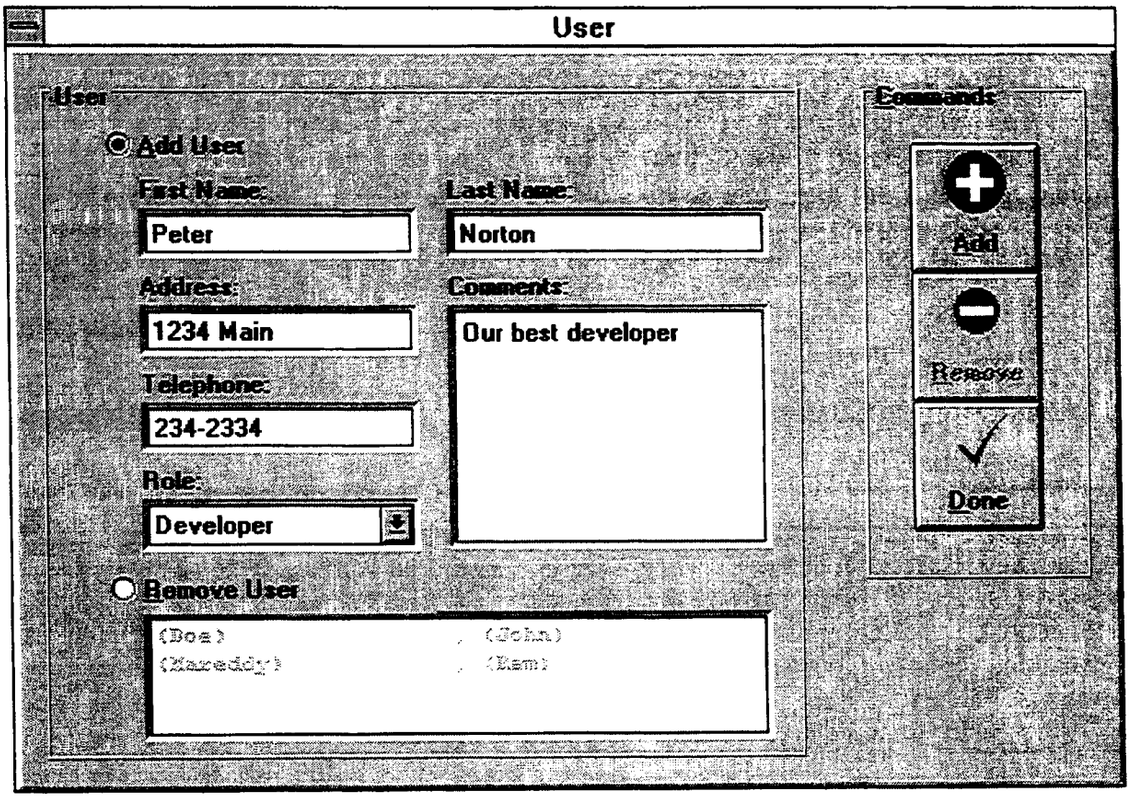

Figure E-15. Administration of Users

To add a user to the system, select the Add User option button. When this option is selected, the Add command button on the right will become active and the Remove command button will become inactive, i.e., it will be grayed out. With the Add User option selected, input the first name, last name, address, telephone, role, and comments in the text boxes and the selection box provided below the option button. Then, to add this user to the system, click on the Add button.

To remove a user from the system, choose the Remove User option button. When this option is selected, the Add command on the right will become inactive and the Remove command button will become active. When the Remove User option button is selected, a list of existing users will be presented in the list box below this option. Select the user that needs to be deleted and click on the Remove command button.

5.6 Role

To add or remove a role, choose the Role command from the Librarian menu. A dialog box similar to Figure E-16 will be presented to the user to facilitate the administration of roles.

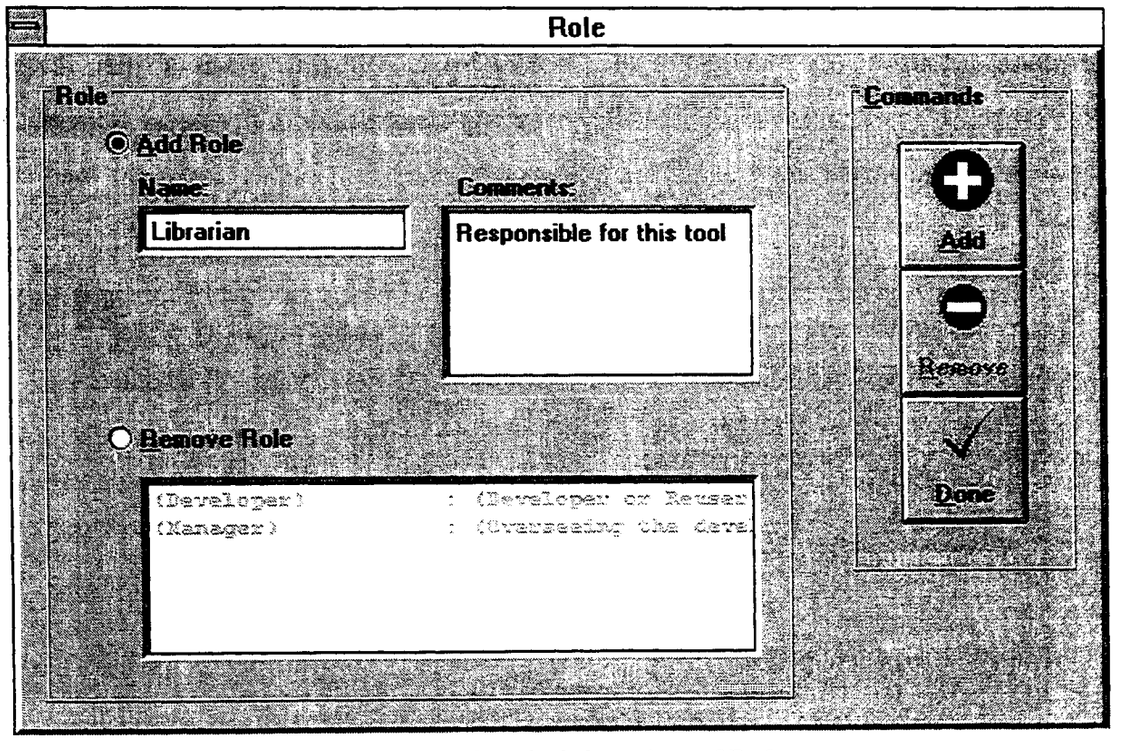

Figure E-16. Administration of Roles

To add a role to the system, select the Add Role option button. When this option is selected, the Add command button on the right will become active and the Remove command button will become inactive, i.e., it will be grayed out. With the Add Role option button selected, input the name of the role and a brief description in the two text boxes provided below the option button. Then, to add this role to the system, click on the Add button.

To remove a role from the system, choose the Remove Role option button. When this option is selected, the Add command button on the right will become inactive and the Remove command button will become active. When the Remove Role option button is selected, a list of the existing roles will be presented in the list box below this option. Select the role that needs to be deleted and click on the Remove command button. Please note here that the role of Librarian cannot be deleted, since someone has to maintain this system. All the users with the deleted role will be temporarily given the Librarian role. These roles should be changed later on.

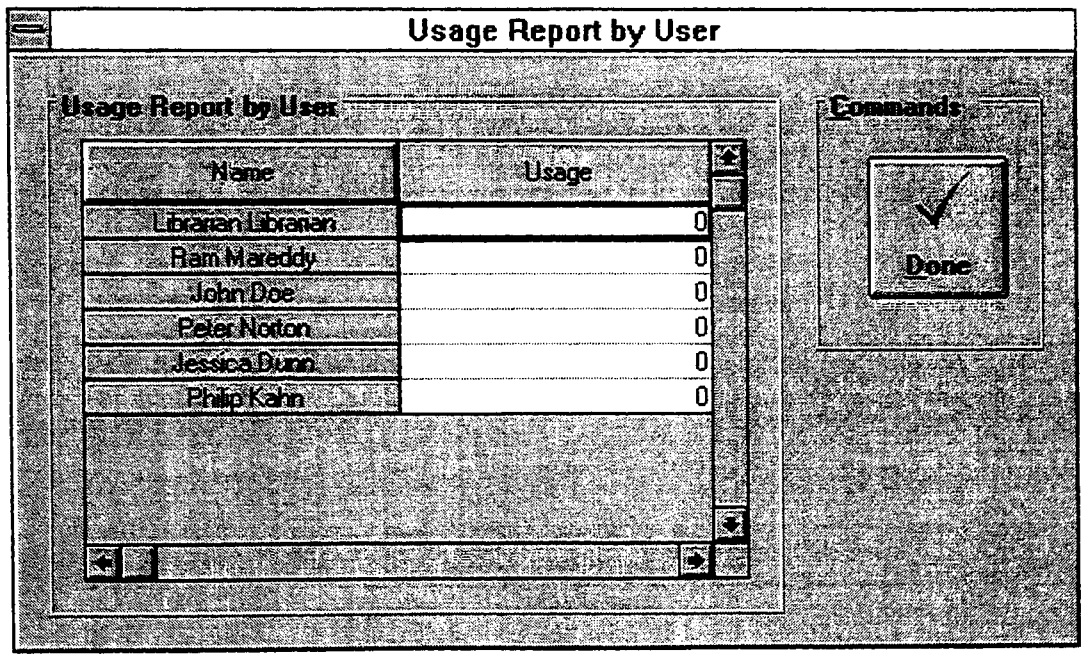

Figure E-17. Usage Report by User

6. Report Generation

Two types of reports can be generated from this tool. The first type deals with the amount of reuse done per user, and the second type deals with amount of reuse done by the artifact.

6.1 Usage Report By User

To obtain the usage report per user, choose the command User Report from the Reports menu. This will generate a report that looks like Figure E-17.

This report gives information about how many functions a user has reused. This information can be used to reward that particular user, or for other organizational purposes.

6.2 Usage Report by Artifact

To generate this report, choose the command Artifact Report from the Reports menu. This will generate a report that looks like Figure E-18.

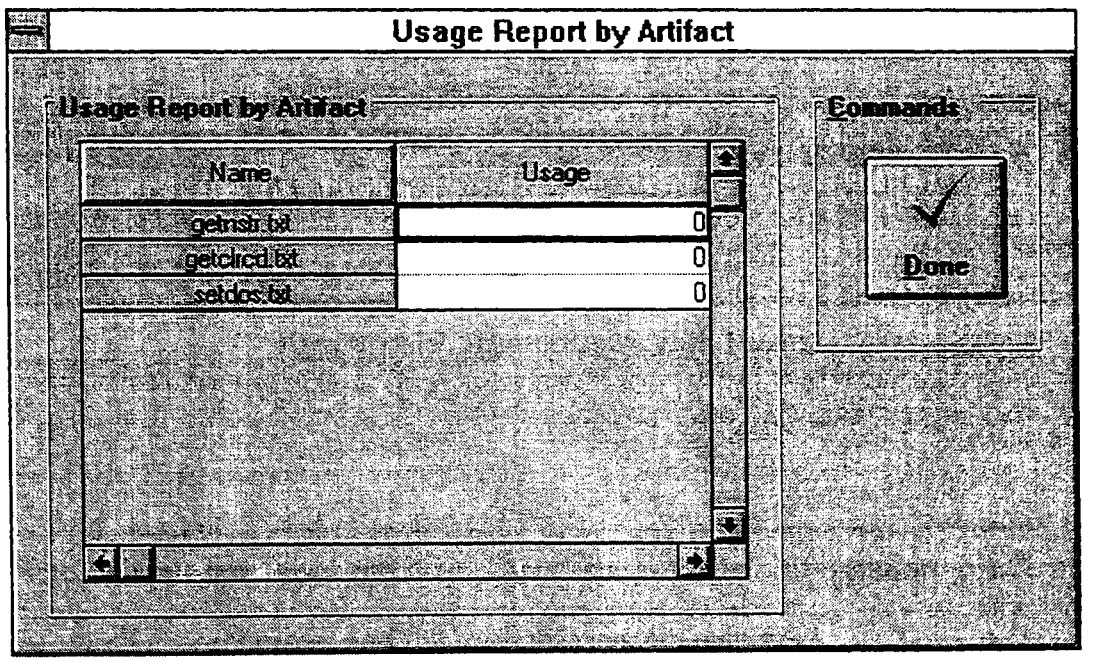

Figure E-18. Usage Report By Artifact

This report lists all the software artifacts listed in the system and the number of times each artifact is reused. This number will help the Librarian in making the decision about removing artifacts due to non-use or low reuse.

7. Help

In the Help menu, an "About..." command is also provided, which gives the version details of the Reuse with Facets tool. This dialog box is depicted in Figure E-19.

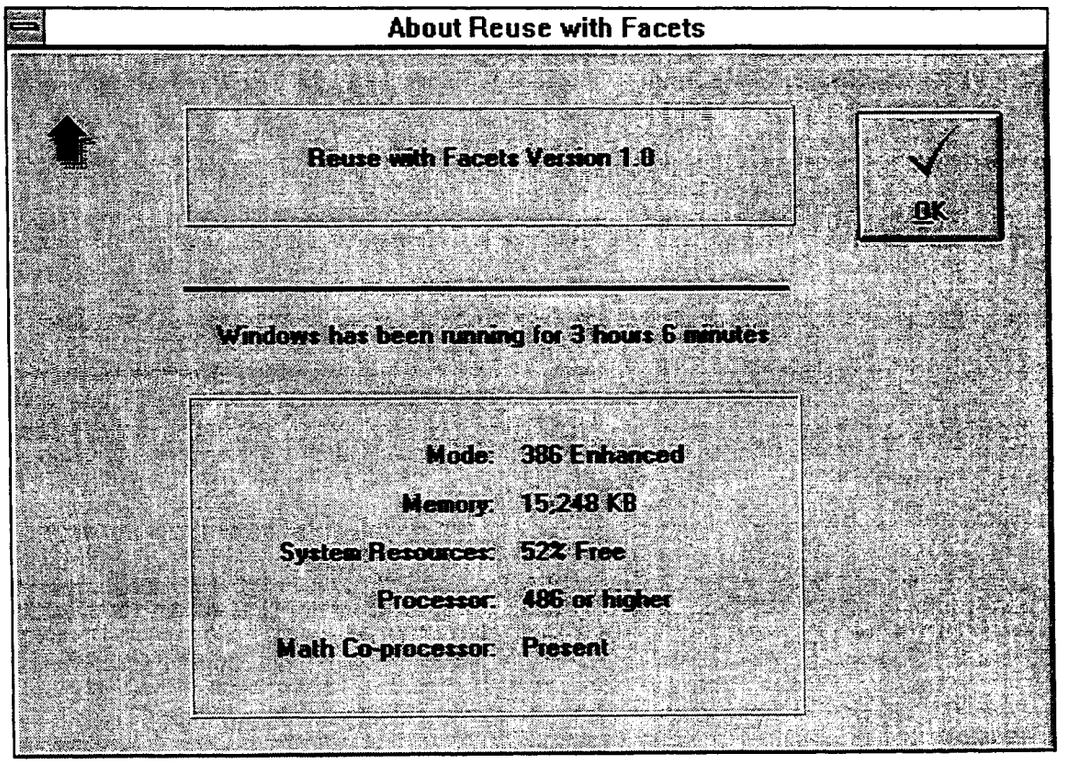

Figure E-19. About Reuse with Facets

The About box also provides the system-related information concerning the processor, math co-processor, and the available system resources.

APPENDIX F

PROGRAM LISTINGS

```
'Make File For The Project
RWFGLOB.BAS
C:\WINDOWS\SYSTEM\SPIN.VBX
C:\WINDOWS\SYSTEM\THREED.VBX
C:\WINDOWS\SYSTEM\SPREAD20.VBX
RWFABOUT.FRM
RWFUSAGE.FRM
RWFROLE.FRM
RWFUSER.FRM
RWFTHES.FRM
RWFVOCAB.FRM
RWFRTRV.FRM
RWFCLASS.FRM
RWFMAIN.FRM
RWFFACET.FRM
RWFDET.FRM
RWFSRCH.FRM
RWFARDEL.FRM
RWFCLREP.FRM
RWFLOGIN.FRM
ProjWinSize=128,386,248,310
ProjWinShow=2
IconForm="frmMain"
Title="RWF"
ExeName="RWF.EXE"

'Initialization File
[Settings]
Maximized=1
Toolbar=1
Statusbar=1
MainWindow=615 720 9740 7825
FontName=MS SANS SERIF
FontBold=0
FontSize=8

[DataSource]
Default=C:\Ramsoft\thesis\program\RWF.MDB

'FrmArtifactDeletion
VERSION 2.00
Begin Form frmArtifactDeletion
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Artifact"
   ClientHeight    =   3255
   ClientLeft      =   -180
   ClientTop       =   2340
   ClientWidth     =   6975
   Height          =   3660
   Left            =   -240
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   ScaleHeight     =   3255
   ScaleWidth      =   6975
   Top             =   1995
   Width           =   7095
   Begin SSPanel Panel3D1
      BevelWidth      =   3
      BorderWidth     =   1
      Height          =   3255
      Left            =   0
      TabIndex        =   0
      Top             =   0
      Width           =   6975
      Begin SSFrame Frame3D1
         Caption         =   "Artifacts"
```

```
      ForeColor       =   &H00000000&
      Height          =   2655
      Left            =   240
      ShadowStyle     =   1  'Raised
      TabIndex        =   4
      Top             =   240
      Width           =   4455
      Begin SSPanel Panel3D3
         AutoSize        =   3  'AutoSize Child To Panel
         BevelOuter      =   1  'Inset
         BevelWidth      =   3
         BorderWidth     =   1
         Height          =   2010
         Left            =   120
         TabIndex        =   5
         Top             =   360
         Width           =   4095
         Begin ListBox lstNamePath
            FontBold        =   -1  'True
            FontItalic      =   0   'False
            FontName        =   "Courier New"
            FontSize        =   8.25
            FontStrikethru  =   0   'False
            FontUnderline   =   0   'False
            Height          =   1920
            Left            =   45
            MultiSelect     =   1  'Simple
            TabIndex        =   6
            Top             =   45
            Width           =   4005
         End
      End
   End
   Begin SSFrame Frame3D2
      Caption         =   "&Commands"
      ForeColor       =   &H00000000&
      Height          =   2655
      Left            =   5040
      ShadowStyle     =   1  'Raised
      TabIndex        =   1
      Top             =   240
      Width           =   1695
      Begin SSCommand cmd3dDone
         AutoSize        =   2  'Adjust Button Size To Picture
         Caption         =   "&Done"
         Height          =   975
         Left            =   360
         Picture         =   RWFARDEL.FRX:0000
         TabIndex        =   3
         Top             =   1320
         Width           =   975
      End
      Begin SSCommand cmd3dRemove
         AutoSize        =   2  'Adjust Button Size To Picture
         Caption         =   "&Remove"
         Height          =   975
         Left            =   360
         Picture         =   RWFARDEL.FRX:0302
         TabIndex        =   2
         Top             =   360
         Width           =   975
      End
   End
End
Option Explicit
```

```
Sub cmd3dDone_Click ()

    Unload Me

End Sub

Sub cmd3dRemove_Click ()

    RemoveArtifact

End Sub

'------------------------------------------------------------------------
'Name: FillBoxes
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This subroutine fills all the box, i.e. listbox with information.
'This can also be called after REMOVE is done.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub FillBoxes ()

    Dim sqryName As String
    Dim sNamePath As String
    Dim dsName As Dynaset


    '------------------
    'Get Name and Group
    '------------------
    sqryName = "SELECT ArtifactFileName, ArtifactPath FROM ARTIFACT"
    Set dsName = gdbRwf.CreateDynaset(sqryName)


    '-----------------------
    'Fill Name+Path List Box
    '-----------------------
    'First clear the list box.
    lstNamePath.Clear

    Do Until dsName.EOF
        sNamePath = "(" + dsName("ArtifactFileName") + ")"
        sNamePath = sNamePath + Space$(20 - Len(sNamePath)) + " : "
        sNamePath = sNamePath + "(" + dsName("ArtifactPath") + ")"
        lstNamePath.AddItem sNamePath
        dsName.MoveNext
    Loop

End Sub

Sub Form_Load ()

    '---------------
    'Center the form
    '---------------
    CenterForm Me

    '----------------------------------
    'Fill boxes on the form with details
    '----------------------------------
```

```
    FillBoxes

End Sub

'------------------------------------------------------------------------
'Name: RemoveArtifact
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function removes one or more artifacts selected in the
'remove list box.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub RemoveArtifact ()

    '---------------------
    'Variable Declarations
    '---------------------
    'Loop counter.
    Dim i As Integer

    'Full string in the remove box.
    Dim sNamePath As String

    'Name of the artifact and path.
    Dim sName As String
    Dim sPath As String

    'Start and end for string parsing.
    Dim nStart As Integer
    Dim nLength As Integer

    'Query to remove the artifact.
    Dim sqryRemoveArtifact As String


    '---------------------------------------
    'All the items selected should be removed
    '---------------------------------------
    For i = 0 To lstNamePath.ListCount - 1
        If lstNamePath.Selected(i) = True Then
            sNamePath = lstNamePath.List(i)

            'Parse the sNamePath into name and path

            'Format of name is (name)   : (path), length of name is position 2 to
            'the character before ).
            nStart = 2
            nLength = InStr(1, sNamePath, ")") - nStart
            sName = Mid$(sNamePath, nStart, nLength)

            'From the second ( to just before the last character.
            'Names are unique by the group.
            nStart = InStr(2, sNamePath, "(") + 1
            nLength = Len(sNamePath) - nStart
            sPath = Mid$(sNamePath, nStart, nLength)

            sqryRemoveArtifact = "DELETE FROM ARTIFACT WHERE ArtifactFileName = '" + sName
      + "AND ArtifactPath = '" + sPath + "'"
            gdbRwf.Execute (sqryRemoveArtifact)
        End If
```

```
    Next


    '------------------
    'Refresh boxes again
    '------------------
    FillBoxes

End Sub

'FrmClassification
VERSION 2.00
Begin Form frmClassification
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Classification"
   ClientHeight    =   4095
   ClientLeft      =   1050
   ClientTop       =   1590
   ClientWidth     =   7320
   Height          =   4500
   Left            =   990
   LinkTopic       =   "Form2"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   ScaleHeight     =   4095
   ScaleWidth      =   7320
   Top             =   1245
   Width           =   7440
   Begin SSPanel Panel3D1
      BevelWidth      =   2
      Height          =   4095
      Left            =   0
      TabIndex        =   0
      Top             =   0
      Width           =   7335
      Begin SSFrame Frame3D3
         Caption         =   "&Commands"
         ForeColor       =   &H00000000&
         Height          =   2535
         Left            =   5520
         ShadowStyle     =   1  'Raised
         TabIndex        =   3
         Top             =   240
         Width           =   1575
         Begin SSCommand cmd3dInsert
            AutoSize        =   2  'Adjust Button Size To Picture
            Caption         =   "&Insert"
            Height          =   975
            Left            =   240
            Picture         =   RWFCLASS.FRX:0000
            TabIndex        =   5
            Top             =   360
            Width           =   1095
         End
         Begin SSCommand cmd3dCancel
            AutoSize        =   2  'Adjust Button Size To Picture
            Caption         =   "&Cancel"
            Height          =   975
            Left            =   240
            Picture         =   RWFCLASS.FRX:0302
            TabIndex        =   4
            Top             =   1320
            Width           =   1095
         End
      End
      Begin SSFrame Frame3D1
         Caption         =   "Facets and Vocabulary"
```

```
        ForeColor       =   &H00000000&
        Height          =   3615
        Left            =   240
        ShadowStyle     =   1  'Raised
        TabIndex        =   1
        Top             =   240
        Width           =   5055
        Begin SpreadSheet sprdInsert
           AllowResize     =   -1  'True
           FontBold        =   -1  'True
           FontItalic      =   0   'False
           FontName        =   "MS Sans Serif"
           FontSize        =   8.25
           FontStrikethru  =   0   'False
           FontUnderline   =   0   'False
           Height          =   3015
           InterfaceDesigner=   RWFCLASS.FRX:0604
           Left            =   240
           MaxCols         =   1
           MaxRows         =   20
           TabIndex        =   2
           Top             =   360
           Width           =   4575
        End
      End
   End
End
Option Explicit
'Row and column numbers.
Const HEADING_ROW_NUMBER = 0
Const FACET_COLUMN_NUMBER = 0
Const VOCABULARY_COLUMN_NUMBER = 1

Sub cmd3dCancel_Click ()

    Unload Me

End Sub

Sub cmd3dInsert_Click ()

    InsertIntoRepository

    Unload Me

End Sub

'--------------------------------------------------------------------------
'Name: FillSpread
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: Spreadsheet in the current form is filled with the requested
'information.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
'
Sub FillSpread ()

    '---------------------
    'Variable Declarations
    '---------------------
```

```
    'Number of facets in the system.
    Dim nNumFacets As Integer

    'Loop counters.
    Dim i As Integer
    Dim j As Integer

    'Number of columns in the spread.
    Const NUM_COLUMNS_IN_SPREAD = 2


    '-------------------------------------
    'Get the number of facets in the system
    '-------------------------------------
    nNumFacets = GetNumFacets()


    '-----------------------------------------
    'Set the maximum number of rows in the spread
    '-----------------------------------------
    sprdInsert.MaxRows = nNumFacets


    '---------------
    'Clear the Spread
    '---------------
    For i = 0 To NUM_COLUMNS_IN_SPREAD
        For j = 1 To nNumFacets
            sprdInsert.Row = j
            sprdInsert.Col = i
            sprdInsert.Text = ""
        Next
    Next


    '--------------------
    'Fill facets in search
    '--------------------
    FillFacetsInInsert


    '-------------------------
    'Fill Vocabulary in search
    '-------------------------
    'Initially put ALL in all the vocabulary.
    For j = 1 To nNumFacets
        sprdInsert.Row = j
        sprdInsert.Col = VOCABULARY_COLUMN_NUMBER
        sprdInsert.Text = "ALL"
    Next j


End Sub

Sub Form_Load ()

    '---------------
    'Center the form
    '---------------
    CenterForm Me

    '----------
    'Set spread
    '----------
    SetSpread
```

```
    '-----------
    'Fill Spread
    '-----------
    FillSpread


End Sub

'------------------------------------------------------------------------
'Name: GetClassificationString
'
'Type: Subroutine
'
'Parameters: sClassification
'
'Returns: None
'
'Description: Classification of the artifact is obtained from the spread,
'after the user has chosen appropriate facets and vocabulary.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub GetClassificationString (sClassification As String)

    '--------------------
    'Variable Declaration
    '--------------------
    'Loop counter.
    Dim i As Integer

    'Selected vocabulary for the facet.
    Dim sVocabularyName As String

    'Id of the vocabulary for the facet.
    Dim nVocabularyId As Integer


    '-----------------------------------------------------------
    'Get Vocabulary Ids and classification for current selection
    '-----------------------------------------------------------
    For i = 1 To sprdInsert.MaxRows
        'Get vocabulary name.
        sprdInsert.Col = VOCABULARY_COLUMN_NUMBER
        sprdInsert.Row = i
        sVocabularyName = sprdInsert.Text

        'Get vocabulary id.
        nVocabularyId = GetVocabularyId(sVocabularyName)

        'Get the classification.
        If i <> 1 Then
            sClassification = sClassification + "," + CStr(nVocabularyId)
        Else
            sClassification = CStr(nVocabularyId)
        End If

    Next i

End Sub

'------------------------------------------------------------------------
'Name: GetFacetString
'
'Type: Subroutine
'
```

```
'Parameters: sFacetString
'
'Returns: sFacetString, facet string is returned in the parameter.
'
'Description: String of facet ids is selected from the spread.
'
'Author: Ram Mareddy
'-------------------------------------------------------------------------
'
Sub GetFacetString (sFacetString As String)

    '--------------------
    'Variable Declaration
    '--------------------
    'Loop counter.
    Dim i As Integer

    'Selected facet.
    Dim sFacetName As String

    'Id of the facet.
    Dim nFacetId As Integer


    '-----------------------------------------------------------
    'Get Facet Ids and classification for current selection
    '-----------------------------------------------------------
    For i = 1 To sprdInsert.MaxRows
        'Get vocabulary name.
        sprdInsert.Col = FACET_COLUMN_NUMBER
        sprdInsert.Row = i
        sFacetName = sprdInsert.Text

        'Get vocabulary id.
        nFacetId = GetFacetId(sFacetName)

        'Get the classification.
        If i <> 1 Then
            sFacetString = sFacetString + "," + CStr(nFacetId)
        Else
            sFacetString = CStr(nFacetId)
        End If

    Next i


End Sub

'-------------------------------------------------------------------------
'Name: InsertIntoRepository
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: Current artifact, with description and classification is put
'into the repository.
'
'Author: Ram Mareddy
'-------------------------------------------------------------------------
'
Sub InsertIntoRepository ()

    '--------------------
    'Variable Declaration
```

```
    '--------------------
    'Classification string.
    Dim sClassification As String
    Dim sFacet As String


    'Loop counter.
    Dim i As Integer

    'Artifacts fitting the classificaion.
    Dim nNumHits As Integer
    Dim sqryArtifactId As String
    Dim dsArtifactId As Dynaset
    Dim sArtifactId As String
    Dim nMaxArtifactId As Integer
    Dim nArtifact As Integer
    Dim nArtifactId As Integer
    Dim tblArtifact As Table


    '-------------------------------------------------------------
    'Get Vocabulary Ids and classification for current selection
    '-------------------------------------------------------------
    GetFacetString sFacet
    GetClassificationString sClassification


    '-----------------------------
    'Inserting into artifact table
    '-----------------------------
    'Get maximum artifact id.
    sqryArtifactId = "SELECT ArtifactId FROM ARTIFACT"
    Set dsArtifactId = gdbRwf.CreateDynaset(sqryArtifactId)

    nMaxArtifactId = 1
    Do Until dsArtifactId.EOF
        sArtifactId = dsArtifactId("ArtifactId")
        If nMaxArtifactId < CInt(sArtifactId) Then
            nMaxArtifactId = CInt(sArtifactId)
        End If
        dsArtifactId.MoveNext
    Loop

    nArtifactId = nMaxArtifactId + 1

    '--------------
    'Add New Record
    '--------------
    Set tblArtifact = gdbRwf.OpenTable("ARTIFACT")

    tblArtifact.AddNew
    tblArtifact("ArtifactId") = nArtifactId
    tblArtifact("FacetIdString") = sFacet
    tblArtifact("VocabularyIdString") = sClassification
    tblArtifact("ArtifactFileName") = gsArtifactFileName
    tblArtifact("ArtifactPath") = gsArtifactFilePath
    tblArtifact("ArtifactSize") = gnArtifactFileLength
    tblArtifact("ArtifactDescription") = gsArtifactDescription
    tblArtifact("ArtifactAuthor") = gsArtifactAuthor
    tblArtifact("ArtifactDateOfInsertion") = Now
    tblArtifact.Update
    tblArtifact.Close


End Sub
```

```
'--------------------------------------------------------------------------------
'Name: SetSpread
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function sets up the spread on the main form.
'
'Author : Ram Mareddy
'--------------------------------------------------------------------------------
'
Sub SetSpread ()


    '------------
    'Declarations
    '------------

    'Row heights and column widths.
    Const HEADING_ROW_HEIGHT = 20
    Const FACET_COLUMN_WIDTH = 17.9
    Const VOCABULARY_COLUMN_WIDTH = 17.9


    '-----------------------------
    'Row Heights and column widths
    '-----------------------------
    'Heading row.
    sprdInsert.RowHeight(HEADING_ROW_NUMBER) = HEADING_ROW_HEIGHT

    'Column widths
    sprdInsert.ColWidth(FACET_COLUMN_NUMBER) = FACET_COLUMN_WIDTH
    sprdInsert.ColWidth(VOCABULARY_COLUMN_NUMBER) = VOCABULARY_COLUMN_WIDTH

    '-------------
    'Column Titles
    '-------------
    sprdInsert.FontSize = gudtRwfIni.nFontSize
    sprdInsert.FontName = gudtRwfIni.sFontName
    sprdInsert.FontBold = gudtRwfIni.bFontBold

    SpreadSetText sprdInsert, FACET_COLUMN_NUMBER, HEADING_ROW_NUMBER, "Facet"
    SpreadSetText sprdInsert, VOCABULARY_COLUMN_NUMBER, HEADING_ROW_NUMBER, "Vocabulary"


    '-----------------
    'Column cell types
    '-----------------
    'frmmain.sprBoard.Col = COLUMN_RETURNS
    'frmmain.sprBoard.Row = SS_ALL_ROWS
    'frmmain.sprBoard.CellType = SS_CELL_TYPE_TIME
    'frmmain.sprBoard.TypeTime24Hour = SS_CELL_TIME_12_HOUR_CLOCK
    'frmmain.sprBoard.TypeTimeSeconds = False
    'frmmain.sprBoard.TypeSpin = True
    'frmmain.sprBoard.TypeHAlign = SS_CELL_H_ALIGN_LEFT
    'frmmain.sprBoard.TypeTimeSeparator = 58
    'frmmain.sprBoard.TypeTimeMin = "000000"
    'frmmain.sprBoard.TypeTimeMax = "235959"


    '------------------
    'Locking the Spread
    '------------------
    'To start with lock the whole spread.
```

```
    'frmmain.sprBoard.Col = SS_ALL_COLUMNS
    'frmmain.sprBoard.Row = SS_ALL_ROWS
    'frmmain.sprBoard.Lock = True


    '------------------
    'Column Appearances
    '------------------
    sprdInsert.Row = SS_ALL_ROWS
    sprdInsert.Col = VOCABULARY_COLUMN_NUMBER
    sprdInsert.TypeHAlign = SS_CELL_H_ALIGN_RIGHT


End Sub

'--------------------------------------------------------------------------
'Name: sprdSearch_Click
'
'Type: Subroutine
'
'Parameters: Col is the column of the selected cell, Row is the row of the
'selected cell.
'
'Returns: None
'
'Description: If the cell is in the vocabulary column, then put a combo box
'with the related vocabulary as contents.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
'
Sub sprdInsert_Click (Col As Long, Row As Long)

    '---------------------
    'Combo with vocabulary
    '---------------------
    If Col = VOCABULARY_COLUMN_NUMBER Then
        FillInsertCombo Col, Row
    End If

End Sub

'--------------------------------------------------------------------------
'Name: sprdSearch_LeaveCell
'
'Type: Subroutine
'
'Parameters: Col is the column of the selected cell, Row is the row of the
'selected cell, NewCol is the column of the cell to which the selection is
'moving to and NewRow is row of the same cell.
'
'Returns: None
'
'Description: If the cell is in the vocabulary column, then take out the
'combo box that is presently in that cell.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
'
Sub sprdInsert_LeaveCell (Col As Long, Row As Long, NewCol As Long, NewRow As Long, Cancel
As Integer)

    '---------------------
    'Variable Declarations
    '---------------------
    'Column which contains the vocabulary terms.
    Const VOCABULARY_COLUMN_NUMBER = 1
```

```
    Const SYNONYM_COLUMN_NUMBER = 2

    '-----------------------
    'Make the cell type text
    '-----------------------
    If (Col = VOCABULARY_COLUMN_NUMBER) Or (Col = SYNONYM_COLUMN_NUMBER) Then
        sprdInsert.CellType = SS_CELL_TYPE_STATIC_TEXT
        sprdInsert.TypeHAlign = SS_CELL_H_ALIGN_RIGHT
    End If


End Sub

'frmUserUsage
VERSION 2.00
Begin Form frmUserUsage
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Usage Report By User"
   ClientHeight    =   4080
   ClientLeft      =   885
   ClientTop       =   1695
   ClientWidth     =   7320
   Height          =   4485
   Left            =   825
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   ScaleHeight     =   4080
   ScaleWidth      =   7320
   Top             =   1350
   Width           =   7440
   Begin SSPanel Panel3D1
      BevelWidth      =   2
      Height          =   4095
      Left            =   0
      TabIndex        =   0
      Top             =   0
      Width           =   7335
      Begin SSFrame Frame3D1
         Caption         =   "Usage Report By User"
         ForeColor       =   &H00000000&
         Height          =   3615
         Left            =   240
         ShadowStyle     =   1  'Raised
         TabIndex        =   2
         Top             =   240
         Width           =   5055
         Begin SpreadSheet sprdUsage
            AllowResize     =   -1  'True
            FontBold        =   -1  'True
            FontItalic      =   0   'False
            FontName        =   "MS Sans Serif"
            FontSize        =   8.25
            FontStrikethru  =   0   'False
            FontUnderline   =   0   'False
            Height          =   3015
            InterfaceDesigner=   RWFCLREP.FRX:0000
            Left            =   240
            MaxCols         =   1
            MaxRows         =   20
            TabIndex        =   3
            Top             =   360
            Width           =   4575
         End
      End
      Begin SSFrame Frame3D3
         Caption         =   "&Commands"
```

```
        ForeColor       =   &H00000000&
        Height          =   1815
        Left            =   5520
        ShadowStyle     =   1  'Raised
        TabIndex        =   1
        Top             =   240
        Width           =   1575
        Begin SSCommand cmd3dDone
           AutoSize        =   2  'Adjust Button Size To Picture
           Caption         =   "&Done"
           Height          =   975
           Left            =   360
           Picture         =   RWFCLREP.FRX:0004
           TabIndex        =   4
           Top             =   480
           Width           =   975
        End
     End
  End
End
Option Explicit
'Row and column numbers.
Const HEADING_ROW_NUMBER = 0
Const NAME_COLUMN_NUMBER = 0
Const USAGE_COLUMN_NUMBER = 1

Sub cmd3dDone_Click ()

    Unload Me

End Sub

'------------------------------------------------------------------------
'Name: FillSpread
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: Spreadsheet in the current form is filled with the requested
'information.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub FillSpread ()

    '---------------------
    'Variable Decalrations
    '---------------------
    'Queries.
    Dim sqryUser As String
    Dim sqryUsage As String

    'Dynasets.
    Dim dsUser As Dynaset
    Dim dsUsage As Dynaset

    'Usage and row
    Dim nSpreadRow As Integer
    Dim nNumRecords As Integer
    Dim nCurrentUserId As Integer


    '------------------------------
```

```
    'Get the name and id of the user
    '-------------------------------
    sqryUser = "SELECT UserId, UserFirstName, UserLastName FROM USERS"
    Set dsUser = gdbRwf.CreateDynaset(sqryUser)

    'Get the record count.
    Do Until dsUser.EOF
        nNumRecords = nNumRecords + 1
        dsUser.MoveNext
    Loop

    sprdUsage.MaxRows = nNumRecords

    nSpreadRow = 1
    dsUser.MoveFirst
    Do Until dsUser.EOF
        'Put name on the spread.
        sprdUsage.Row = nSpreadRow
        sprdUsage.Col = NAME_COLUMN_NUMBER
        sprdUsage.Text = dsUser("UserFirstName") + " " + dsUser("UserLastName")

        'Get the number of events.
        nCurrentUserId = dsUser("UserId")
        sqryUsage = "SELECT UserId FROM EVENT WHERE UserId = " + CStr(nCurrentUserId)
        Set dsUsage = gdbRwf.CreateDynaset(sqryUsage)

        nNumRecords = 0
        Do Until dsUsage.EOF
            nNumRecords = nNumRecords + 1
            dsUsage.MoveNext
        Loop

        'Put the usage.
        sprdUsage.Row = nSpreadRow
        sprdUsage.Col = USAGE_COLUMN_NUMBER
        sprdUsage.Text = nNumRecords

        dsUser.MoveNext
        nSpreadRow = nSpreadRow + 1

    Loop


End Sub

Sub Form_Load ()

    '---------------
    'Center the form
    '---------------
    CenterForm Me

    '----------
    'Set spread
    '----------
    SetSpread


    '-----------
    'Fill Spread
    '-----------
    FillSpread


End Sub

'------------------------------------------------------------------------------
```

```
'Name: SetSpread
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function sets up the spread on the usage report form.
'
'Author : Ram Mareddy
'--------------------------------------------------------------------------------
'
Sub SetSpread ()


    '------------
    'Declarations
    '------------

    'Row heights and column widths.
    Const HEADING_ROW_HEIGHT = 20
    Const NAME_COLUMN_WIDTH = 17.9
    Const USAGE_COLUMN_WIDTH = 17.9


    '-----------------------------
    'Row Heights and column widths
    '-----------------------------
    'Heading row.
    sprdUsage.RowHeight(HEADING_ROW_NUMBER) = HEADING_ROW_HEIGHT

    'Column widths
    sprdUsage.ColWidth(NAME_COLUMN_NUMBER) = NAME_COLUMN_WIDTH
    sprdUsage.ColWidth(USAGE_COLUMN_NUMBER) = USAGE_COLUMN_WIDTH

    '-------------
    'Column Titles
    '-------------
    sprdUsage.FontSize = gudtRwfIni.nFontSize
    sprdUsage.FontName = gudtRwfIni.sFontName
    sprdUsage.FontBold = gudtRwfIni.bFontBold

    SpreadSetText sprdUsage, NAME_COLUMN_NUMBER, HEADING_ROW_NUMBER, "Name"
    SpreadSetText sprdUsage, USAGE_COLUMN_NUMBER, HEADING_ROW_NUMBER, "Usage"


    '------------------
    'Column Appearances
    '------------------
    sprdUsage.Row = SS_ALL_ROWS
    sprdUsage.Col = USAGE_COLUMN_NUMBER
    sprdUsage.TypeHAlign = SS_CELL_H_ALIGN_RIGHT


End Sub

'frmDetails
VERSION 2.00
Begin Form frmDetails
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Artifact Details"
   ClientHeight    =   4695
   ClientLeft      =   1035
   ClientTop       =   1530
   ClientWidth     =   6975
   Height          =   5100
```

```
Left            =   975
LinkTopic       =   "Form1"
MaxButton       =   0   'False
MinButton       =   0   'False
ScaleHeight     =   4695
ScaleWidth      =   6975
Top             =   1185
Width           =   7095
Begin SSPanel Panel3D1
   BevelWidth      =   2
   Height          =   4695
   Left            =   0
   TabIndex        =   6
   Top             =   0
   Width           =   6975
   Begin SSFrame Frame3D3
      Caption         =   "Commands"
      ForeColor       =   &H00000000&
      Height          =   2535
      Left            =   5040
      ShadowStyle     =   1  'Raised
      TabIndex        =   17
      Top             =   240
      Width           =   1575
      Begin SSCommand cmd3dProceed
         AutoSize        =   2  'Adjust Button Size To Picture
         Caption         =   "&Proceed..."
         Height          =   975
         Left            =   240
         Picture         =   RWFDET.FRX:0000
         TabIndex        =   4
         Top             =   360
         Width           =   1095
      End
      Begin SSCommand cmd3dCancel
         AutoSize        =   2  'Adjust Button Size To Picture
         Caption         =   "&Cancel"
         Height          =   975
         Left            =   240
         Picture         =   RWFDET.FRX:0302
         TabIndex        =   5
         Top             =   1320
         Width           =   1095
      End
   End
   Begin SSFrame Frame3D2
      Caption         =   "General"
      ForeColor       =   &H00000000&
      Height          =   2295
      Left            =   360
      ShadowStyle     =   1  'Raised
      TabIndex        =   8
      Top             =   2040
      Width           =   4335
      Begin SSPanel Panel3D5
         AutoSize        =   3  'AutoSize Child To Panel
         BevelOuter      =   1  'Inset
         BevelWidth      =   2
         Height          =   1215
         Left            =   1440
         TabIndex        =   16
         Top             =   840
         Width           =   2655
         Begin TextBox txtDescription
            Height          =   1155
            Left            =   30
            MultiLine       =   -1  'True
```

```
      TabIndex        =   3
      Top             =   30
      Width           =   2595
   End
End
Begin SSPanel Panel3D4
   AutoSize        =   3  'AutoSize Child To Panel
   BevelOuter      =   1  'Inset
   BevelWidth      =   2
   Height          =   375
   Left            =   1440
   TabIndex        =   14
   Top             =   240
   Width           =   2655
   Begin TextBox txtAuthor
      Height          =   315
      Left            =   30
      TabIndex        =   2
      Top             =   30
      Width           =   2595
   End
End
Begin Label Label4
   BackColor       =   &H00C0C0C0&
   Caption         =   "&Description:"
   Height          =   255
   Left            =   240
   TabIndex        =   15
   Top             =   960
   Width           =   1095
End
Begin Label Label3
   BackColor       =   &H00C0C0C0&
   Caption         =   "&Author:"
   Height          =   255
   Left            =   240
   TabIndex        =   13
   Top             =   360
   Width           =   735
End
End
Begin SSFrame Frame3D1
   Caption         =   "Location"
   ForeColor       =   &H00000000&
   Height          =   1455
   Left            =   360
   ShadowStyle     =   1  'Raised
   TabIndex        =   7
   Top             =   240
   Width           =   4335
   Begin SSPanel Panel3D3
      AutoSize        =   3  'AutoSize Child To Panel
      BevelOuter      =   1  'Inset
      BevelWidth      =   2
      Height          =   375
      Left            =   1440
      TabIndex        =   12
      Top             =   840
      Width           =   2655
      Begin TextBox txtFilePath
         Height          =   315
         Left            =   30
         TabIndex        =   1
         Top             =   30
         Width           =   2595
      End
   End
```

```
        Begin SSPanel Panel3D2
           AutoSize        =   3  'AutoSize Child To Panel
           BevelOuter      =   1  'Inset
           BevelWidth      =   2
           Height          =   375
           Left            =   1440
           TabIndex        =   10
           Top             =   240
           Width           =   2655
           Begin TextBox txtFileName
              Height          =   315
              Left            =   30
              TabIndex        =   0
              Top             =   30
              Width           =   2595
           End
        End
        Begin Label Label2
           BackColor       =   &H00C0C0C0&
           Caption         =   "File &Path:"
           Height          =   255
           Left            =   240
           TabIndex        =   11
           Top             =   960
           Width           =   855
        End
        Begin Label Label1
           BackColor       =   &H00C0C0C0&
           Caption         =   "File &Name:"
           Height          =   255
           Left            =   240
           TabIndex        =   9
           Top             =   360
           Width           =   1695
        End
     End
  End
End
Option Explicit

'--------------------------------------------------------------------------------
'Name: CheckArtifact
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: The physical existence of the artifact is checked in this
'routine. Error message is displayed, if the artifact does not exist.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------------
'
Sub CheckArtifact ()

    '---------------------
    'Variable Declarations
    '---------------------
    Dim sFile As String

    'Message box variables.
    Dim sMessage As String
    Dim sTitle As String

    On Error Resume Next
```

```
    '---------------
    'Get File Length
    '---------------
    sFile = txtFilePath + txtFileName
    gnArtifactFileLength = FileLen(sFile)

    If Err Then
        sMessage = "File you specified does not exist. Please check and try again."
        sTitle = "File Location Error"
        MsgBox sMessage, MB_OK + MB_ICONEXCLAMATION, sTitle

    Else
        '-------------------------
        'Artifact Physical Details
        '-------------------------
        gsArtifactFileName = txtFileName.Text
        gsArtifactFilePath = txtFilePath.Text
        gsArtifactAuthor = txtAuthor.Text
        gsArtifactDescription = txtDescription.Text

        frmClassification.Show Modal
    End If

End Sub

Sub cmd3dCancel_Click ()

    Unload Me

End Sub

Sub cmd3dProceed_Click ()

    CheckArtifact

    Unload Me

End Sub

Sub Form_Load ()

    '---------------
    'Center the form
    '---------------
    CenterForm Me

End Sub

'frmFacets
VERSION 2.00
Begin Form frmFacets
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Facets"
   ClientHeight    =   5175
   ClientLeft      =   615
   ClientTop       =   3015
   ClientWidth     =   8295
   Height          =   5580
   Left            =   555
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   ScaleHeight     =   5175
   ScaleWidth      =   8295
   Top             =   2670
   Width           =   8415
```

```
Begin SSPanel Panel3D1
   BevelWidth      =   3
   BorderWidth     =   1
   Height          =   5175
   Left            =   0
   TabIndex        =   8
   Top             =   0
   Width           =   8295
   Begin SSFrame Frame3D1
      Caption         =   "Facets"
      ForeColor       =   &H00000000&
      Height          =   4695
      Left            =   240
      ShadowStyle     =   1  'Raised
      TabIndex        =   10
      Top             =   240
      Width           =   5655
      Begin SSPanel Panel3D4
         AutoSize        =   3  'AutoSize Child To Panel
         BevelOuter      =   1  'Inset
         BevelWidth      =   3
         BorderWidth     =   1
         Height          =   1335
         Left            =   3000
         TabIndex        =   13
         Top             =   960
         Width           =   2415
         Begin TextBox txtComments
            Height          =   1245
            Left            =   45
            MultiLine       =   -1  'True
            TabIndex        =   2
            Top             =   45
            Width           =   2325
         End
      End
      Begin SSOption opt3dAddFacet
         Caption         =   "&Add Facet"
         Height          =   255
         Left            =   480
         TabIndex        =   0
         Top             =   360
         Width           =   1215
      End
      Begin SSPanel Panel3D2
         AutoSize        =   3  'AutoSize Child To Panel
         BevelOuter      =   1  'Inset
         BevelWidth      =   3
         BorderWidth     =   1
         Height          =   375
         Left            =   720
         TabIndex        =   12
         Top             =   960
         Width           =   2055
         Begin TextBox txtName
            Height          =   285
            Left            =   45
            TabIndex        =   1
            Top             =   45
            Width           =   1965
         End
      End
      Begin SSOption opt3dRemoveFacet
         Caption         =   "&Remove Facet"
         Height          =   375
         Left            =   480
         TabIndex        =   3
```

```
      TabStop          =   0   'False
      Top              =   2520
      Width            =   1815
   End
   Begin SSPanel Panel3D3
      AutoSize         =   3  'AutoSize Child To Panel
      BevelOuter       =   1  'Inset
      BevelWidth       =   3
      BorderWidth      =   1
      Height           =   1380
      Left             =   720
      TabIndex         =   11
      Top              =   3000
      Width            =   4695
      Begin ListBox lstNameComments
         FontBold         =   -1  'True
         FontItalic       =   0   'False
         FontName         =   "Courier New"
         FontSize         =   8.25
         FontStrikethru   =   0   'False
         FontUnderline    =   0   'False
         Height           =   1290
         Left             =   45
         MultiSelect      =   1  'Simple
         TabIndex         =   4
         Top              =   45
         Width            =   4605
      End
   End
   Begin Label Label2
      BackColor        =   &H00C0C0C0&
      Caption          =   "Comments:"
      Height           =   255
      Left             =   3120
      TabIndex         =   15
      Top              =   720
      Width            =   1095
   End
   Begin Label Label1
      BackColor        =   &H00C0C0C0&
      Caption          =   "Name:"
      Height           =   255
      Left             =   720
      TabIndex         =   14
      Top              =   720
      Width            =   855
   End
End
Begin SSFrame Frame3D2
   Caption          =   "&Commands"
   ForeColor        =   &H00000000&
   Height           =   3735
   Left             =   6360
   ShadowStyle      =   1  'Raised
   TabIndex         =   9
   Top              =   240
   Width            =   1695
   Begin SSCommand cmd3dDone
      AutoSize         =   2  'Adjust Button Size To Picture
      Caption          =   "&Done"
      Height           =   975
      Left             =   360
      Picture          =   RWFFACET.FRX:0000
      TabIndex         =   7
      Top              =   2400
      Width            =   975
   End
```

```
        Begin SSCommand cmd3dRemove
           AutoSize        =   2  'Adjust Button Size To Picture
           Caption         =   "&Remove"
           Height          =   975
           Left            =   360
           Picture         =   RWFFACET.FRX:0302
           TabIndex        =   6
           Top             =   1440
           Width           =   975
        End
        Begin SSCommand cmd3dAdd
           AutoSize        =   2  'Adjust Button Size To Picture
           Caption         =   "&Add"
           Height          =   975
           Left            =   360
           Picture         =   RWFFACET.FRX:0604
           TabIndex        =   5
           Top             =   480
           Width           =   975
        End
     End
  End
End
Option Explicit

'------------------------------------------------------------------------
'Name: AddFacet
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This routine adds a facet to the database.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub AddFacet ()

    '--------------------
    'Variable Declaration
    '--------------------
    'Facet information.
    Dim sFacetName As String
    Dim sFacetComments As String

    'Qurey to obtain the facet ids.
    Dim sqryFacetId As String
    Dim sFacetId As String
    Dim nMaxFacetId As Integer
    Dim nFacetId As Integer

    'Dynaset containing the facet ids.
    Dim dsFacetId As Dynaset

    'Message box variables.
    Dim sMessage As String
    Dim sTitle As String
    Dim nResponse As Integer

    Dim dsUser As Dynaset
    Dim tblFacet As table

    '--------------
    'Check Text Box
```

```
    '--------------
    If (txtName.Text = "") Then
        sMessage = "Name box is empty. Null user can not be added."
        sTitle = "Add Facet"
        nResponse = MsgBox(sMessage, MB_OK + MB_ICONEXCLAMATION, sTitle)
        Exit Sub
    End If


    '-----------------------------------
    'Facet and the description to be added
    '-----------------------------------
    sFacetName = txtName.Text
    sFacetComments = txtComments.Text


    '------------------
    'Get the maximum-id
    '------------------
    sqryFacetId = "SELECT FacetId FROM FACET"
    Set dsFacetId = gdbRwf.CreateDynaset(sqryFacetId)

    Do Until dsFacetId.EOF
        sFacetId = dsFacetId("FacetId")
        If nMaxFacetId < CInt(sFacetId) Then
            nMaxFacetId = CInt(sFacetId)
        End If
        dsFacetId.MoveNext
    Loop

    nFacetId = nMaxFacetId + 1


    '--------------
    'Add New Record
    '--------------
    Set tblFacet = gdbRwf.OpenTable("FACET")

    tblFacet.AddNew
    tblFacet("FacetName") = sFacetName
    tblFacet("FacetComments") = sFacetComments
    tblFacet("FacetId") = nFacetId
    tblFacet.Update
    tblFacet.Close

End Sub

Sub AdjustClassification ()

End Sub

Sub cmd3dAdd_Click ()


    '--------------------
    'Add User to Database
    '--------------------
    AddFacet

    '--------------------
    'Update all the boxes
    '--------------------
    FillBoxes

    '---------------------
    'Set focus to Name Box
    '---------------------
```

```
    txtName.SetFocus


End Sub

Sub cmd3dDone_Click ()

    '---------------------
    'Variable Declarations
    '---------------------
    'Message box variables.
    Dim sMessage As String
    Dim sTitle As String
    Dim nResponse As Integer

    'Check the Text Box to see the user is mistakenly closing.
    If txtName.Text <> "" Then

        sMessage = "Name box is not empty, may be you are trying to add a facet."
        sMessage = sMessage + "  Do you want to go back?"
        sTitle = "Add Facet Message"

        nResponse = MsgBox(sMessage, MB_YESNO, sTitle)

        'if user wants to go back, exit this subroutine.
        If nResponse = IDYES Then
            'Give focus to text box.
            EnableAddFacet
            txtName.SetFocus

            Exit Sub
        End If

      End If


    Unload Me

End Sub

Sub cmd3dRemove_Click ()

    RemoveFacet

End Sub

'--------------------------------------------------------------------------
'Name: EnableAddFacet
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function enables all the add facet boxes and buttons,
'where as the EnableRemoveFacet does the vice versa.
'--------------------------------------------------------------------------
'
Sub EnableAddFacet ()

    '------
    'Enable
    '------
    txtName.Enabled = True
    txtComments.Enabled = True
    cmd3dAdd.Enabled = True
```

```
    '-------
    'Disable
    '-------
    opt3dRemoveFacet.Value = False
    lstNameComments.Enabled = False
    cmd3dRemove.Enabled = False

End Sub

'------------------------------------------------------------------------
'Name: EnableRemoveFacet
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function enables all the remove facet boxes and buttons,
'where as the EnableAddFacet does the vice versa.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub EnableRemoveFacet ()

    '------
    'Enable
    '------
    lstNameComments.Enabled = True
    cmd3dRemove.Enabled = True

    '-------
    'Disable
    '-------
    opt3dAddFacet.Value = False
    txtName.Enabled = False
    txtComments.Enabled = False
    cmd3dAdd.Enabled = False

End Sub

'------------------------------------------------------------------------
'Name: FillBoxes
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This subroutine fills all the boxes, i.e. text box, listbox
'with information. This can also be called after ADD or REMOVE is done.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub FillBoxes ()

    Dim sqryName As String
    Dim sNameDescription As String
    Dim dsName As Dynaset


    '------------------
    'Get Name and Group
```

```
    '-----------------
    sqryName = "SELECT FacetName, FacetComments FROM FACET"
    Set dsName = gdbRwf.CreateDynaset(sqryName)

    '--------------
    'Empty Name Box
    '--------------
    txtName.Text = ""
    txtComments.Text = ""


    '------------------------------
    'Fill Name+Description List Box
    '------------------------------
    'First clear the list box.
    lstNameComments.Clear

    Do Until dsName.EOF
        sNameDescription = "(" + dsName("FacetName") + ")"
        sNameDescription = sNameDescription + Space$(20 - Len(sNameDescription)) + " : "
        sNameDescription = sNameDescription + "(" + dsName("FacetComments") + ")"
        lstNameComments.AddItem sNameDescription
        dsName.MoveNext
    Loop

End Sub

Sub Form_Load ()

    '---------------
    'Center the form
    '---------------
    CenterForm Me

    '-----------------------------------
    'Fill boxes on the form with details
    '-----------------------------------
    FillBoxes

End Sub

Sub opt3dAddFacet_Click (Value As Integer)

    EnableAddFacet

    '---------------------
    'Set focus to Name Box
    '---------------------
    txtName.SetFocus

End Sub

Sub opt3dRemoveFacet_Click (Value As Integer)

    EnableRemoveFacet

End Sub

'-------------------------------------------------------------------------
'Name: RemoveFacet
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
```

```
'Description: This function removes one or more users selected in the
'remove list box.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
'
Sub RemoveFacet ()

    '--------------------
    'Variable Declarations
    '--------------------
    'Loop counter.
    Dim i As Integer

    'Full string in the remove box.
    Dim sNameComments As String

    'Name of the facet.
    Dim sName As String

    'Start and end for string parsing.
    Dim nStart As Integer
    Dim nLength As Integer

    'Query to remove the facet.
    Dim sqryRemoveFacet As String


    '---------------------------------------
    'All the items selected should be removed
    '---------------------------------------
    For i = 0 To lstNameComments.ListCount - 1
        If lstNameComments.Selected(i) = True Then
            sNameComments = lstNameComments.List(i)

            'Parse the sNameComments into name and comments

            'Format of name is (name)  : (comments), length of name is position 2 to
            'the character before ).
            nStart = 2
            nLength = InStr(1, sNameComments, ")") - nStart
            sName = Mid$(sNameComments, nStart, nLength)

            sqryRemoveFacet = "DELETE FROM FACET WHERE FacetName = '" + sName + "'"
            gdbRwf.Execute (sqryRemoveFacet)
        End If
    Next


    '--------------------------------------------------
    'Adjust the classification after the facet is removed
    '--------------------------------------------------
    AdjustClassification


    '------------------
    'Refresh boxes again
    '------------------
    FillBoxes

End Sub

'frmLogin
VERSION 2.00
Begin Form frmLogin
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Login"
```

```
ClientHeight     =   3135
ClientLeft       =   1560
ClientTop        =   2250
ClientWidth      =   4800
Height           =   3540
Left             =   1500
LinkTopic        =   "Form1"
MaxButton        =   0   'False
MinButton        =   0   'False
ScaleHeight      =   3135
ScaleWidth       =   4800
Top              =   1905
Width            =   4920
Begin SSPanel Panel3D1
   BevelWidth       =   3
   BorderWidth      =   1
   Height           =   3135
   Left             =   0
   TabIndex         =   0
   Top              =   0
   Width            =   4815
   Begin SSFrame Frame3D2
      Caption          =   "Commands"
      ForeColor        =   &H00000000&
      Height           =   2655
      Left             =   2880
      ShadowStyle      =   1  'Raised
      TabIndex         =   7
      Top              =   240
      Width            =   1695
      Begin SSCommand cmd3dCancel
         AutoSize         =   2  'Adjust Button Size To Picture
         Caption          =   "&Cancel"
         Height           =   975
         Left             =   360
         Picture          =   RWFLOGIN.FRX:0000
         TabIndex         =   1
         Top              =   1320
         Width            =   975
      End
      Begin SSCommand cmd3dOk
         AutoSize         =   2  'Adjust Button Size To Picture
         Caption          =   "&OK"
         Height           =   975
         Left             =   360
         Picture          =   RWFLOGIN.FRX:0302
         TabIndex         =   2
         Top              =   360
         Width            =   975
      End
   End
   Begin SSFrame Frame3D1
      Caption          =   "Login"
      ForeColor        =   &H00000000&
      Height           =   1695
      Left             =   240
      ShadowStyle      =   1  'Raised
      TabIndex         =   3
      Top              =   240
      Width            =   2295
      Begin SSPanel Panel3D2
         AutoSize         =   3  'AutoSize Child To Panel
         BevelOuter       =   1  'Inset
         BevelWidth       =   3
         BorderWidth      =   1
         Height           =   390
         Left             =   120
```

```
            TabIndex        =   5
            Top             =   1080
            Width           =   1935
            Begin ComboBox cboUser
               Height          =   300
               Left            =   45
               Style           =   2  'Dropdown List
               TabIndex        =   6
               Top             =   45
               Width           =   1845
            End
         End
         Begin Label Label2
            BackColor       =   &H00C0C0C0&
            Caption         =   "Please login by choosing your name:"
            Height          =   495
            Left            =   120
            TabIndex        =   4
            Top             =   360
            Width           =   1935
         End
      End
   End
End
Option Explicit

Sub cmd3dCancel_Click ()

    End

End Sub

Sub cmd3dOk_Click ()

    '---------------------
    'Variable Declarations
    '---------------------
    Dim sUser As String
    Dim nComma As Integer

    'Get current selection of user.
    sUser = cboUser.Text

    'Parse the user name into first and last names.
    nComma = InStr(1, sUser, ",")
    gsLoginLastName = Mid$(sUser, 1, nComma - 1)
    gsLoginFirstName = Mid$(sUser, nComma + 2)

    Unload Me

End Sub

'--------------------------------------------------------------------------
'Name: FillCombo
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: Users in the system are added to a combo box.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
'
Sub FillCombo ()
```

```
    '--------------------
    'Variable Declarations
    '--------------------
    'Query strings
    Dim sqryUser As String
    Dim sUser As String

    'Dynasets.
    Dim dsUser As Dynaset

    'Combo box index.
    Dim nListIndex As Integer


    '-------------------------
    'Get FirstName and LastName
    '-------------------------
    sqryUser = "SELECT UserFirstName, UserLastName FROM USERS"
    Set dsUser = gdbRwf.CreateDynaset(sqryUser)


    '------------------
    'Fill name Combo Box
    '------------------
    'Clear the combo box.
    cboUser.Clear

    Do Until dsUser.EOF
        sUser = dsUser("UserLastName") + ", " + dsUser("UserFirstName")
        cboUser.AddItem sUser
        dsUser.MoveNext
    Loop

    'First time list index will be -1, then change it to 0.
    If nListIndex = -1 Then nListIndex = 0
    cboUser.ListIndex = nListIndex

End Sub

Sub Form_Load ()

    'Put the user names in the combo.
    FillCombo

End Sub

`frmMain
VERSION 2.00
Begin Form frmMain
   Caption         =   "Reuse With Facets"
   ClientHeight    =   4020
   ClientLeft      =   870
   ClientTop       =   1965
   ClientWidth     =   7770
   Height          =   4710
   Icon            =   RWFMAIN.FRX:0000
   Left            =   810
   LinkTopic       =   "Form1"
   Picture         =   RWFMAIN.FRX:0302
   ScaleHeight     =   4020
   ScaleWidth      =   7770
   Top             =   1335
   Width           =   7890
   Begin PictureBox picStatusbar
      Align           =   2  'Align Bottom
      Height          =   375
```

```
   Left            =   0
   ScaleHeight     =   345
   ScaleWidth      =   7740
   TabIndex        =   6
   Top             =   3645
   Width           =   7770
   Begin SSPanel pnl3dStatusbar
      Height          =   375
      Left            =   0
      TabIndex        =   7
      Top             =   0
      Width           =   5895
   End
End
Begin PictureBox picToolbar
   Align           =   1  'Align Top
   Height          =   615
   Left            =   0
   ScaleHeight     =   585
   ScaleWidth      =   7740
   TabIndex        =   0
   Top             =   0
   Width           =   7770
   Begin SSPanel pnl3dToolbar
      Height          =   615
      Left            =   0
      TabIndex        =   1
      Top             =   0
      Width           =   8655
      Begin SSCommand cmdInsert
         AutoSize        =   2  'Adjust Button Size To Picture
         Height          =   600
         Left            =   1680
         Picture         =   RWFMAIN.FRX:0604
         TabIndex        =   5
         Top             =   0
         Width           =   600
      End
      Begin SSCommand cmdExit
         AutoSize        =   2  'Adjust Button Size To Picture
         Height          =   600
         Left            =   240
         Picture         =   RWFMAIN.FRX:0906
         TabIndex        =   4
         Top             =   0
         Width           =   600
      End
      Begin SSCommand cmdSearch
         AutoSize        =   2  'Adjust Button Size To Picture
         Height          =   600
         Left            =   1110
         Picture         =   RWFMAIN.FRX:0C08
         TabIndex        =   3
         Top             =   0
         Width           =   600
      End
      Begin SSCommand cmdHelp
         AutoSize        =   2  'Adjust Button Size To Picture
         Height          =   600
         Left            =   2520
         Picture         =   RWFMAIN.FRX:0F0A
         TabIndex        =   2
         Top             =   0
         Width           =   600
      End
   End
End
```

```
  Begin Menu mnu_User
     Caption         =   "&User"
     Begin Menu mnuUserSearch
        Caption          =   "&Search..."
     End
     Begin Menu mnuUserInsert
        Caption          =   "&Insert..."
     End
     Begin Menu UserSeparator1
        Caption          =   "-"
     End
     Begin Menu mnuExit
        Caption          =   "&Exit"
     End
  End
  Begin Menu mnu_Librarian
     Caption         =   "&Librarian"
     Begin Menu mnuLibrarianFacets
        Caption          =   "&Facets..."
     End
     Begin Menu mnuLibrarianVocabulary
        Caption          =   "&Vocabulary..."
     End
     Begin Menu mnuArtefacts
        Caption          =   "&Artifacts..."
     End
     Begin Menu mnuThesaurus
        Caption          =   "&Thesaurus..."
     End
     Begin Menu separator
        Caption          =   "-"
     End
     Begin Menu mnuUser
        Caption          =   "&User..."
     End
     Begin Menu mnuRole
        Caption          =   "&Role..."
     End
  End
  Begin Menu mnu_Reports
     Caption         =   "&Reports"
     Begin Menu mnuReportsUser
        Caption          =   "&User Usage Report..."
     End
     Begin Menu mnuReportsArtifact
        Caption          =   "&Artifact Usage Report..."
     End
  End
  Begin Menu mnu_Help
     Caption         =   "&Help"
     Begin Menu mnuHelpAbout
        Caption          =   "&About Reuse With Facets..."
     End
  End
End
Option Explicit

Sub cmdExit_Click ()

    mnuExit_Click

End Sub

Sub cmdExit_MouseMove (Button As Integer, Shift As Integer, X As Single, Y As Single)

    pnl3dStatusbar.Caption = "Click to exit the program."
```

```
End Sub
Sub cmdHelp_Click ()
    mnuHelpAbout_Click
End Sub
Sub cmdHelp_MouseMove (Button As Integer, Shift As Integer, X As Single, Y As Single)
    pnl3dStatusbar.Caption = "Click to start the help system."
End Sub
Sub cmdInsert_Click ()
    mnuUserInsert_Click
End Sub
Sub cmdInsert_MouseMove (Button As Integer, Shift As Integer, X As Single, Y As Single)
    pnl3dStatusbar.Caption = "Click to insert into the repository."
End Sub
Sub cmdSearch_Click ()
    mnuUserSearch_Click
End Sub
Sub cmdSearch_MouseMove (Button As Integer, Shift As Integer, X As Single, Y As Single)
    pnl3dStatusbar.Caption = "Click to search the repository."
End Sub
'------------------------------------------------------------------------------
'Name: Main Form_Load
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: Main window is set and maximized.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------------
'
Sub Form_Load ()

    '--------------------------------
    'Put all the details on the screen
    '--------------------------------
    'Display the main screen according to the seetings in ini file.
    SetMainWindow Me


End Sub
Sub mnuArtefacts_Click ()
    frmArtifactDeletion.Show Modal
```

```
End Sub
Sub mnuExit_Click ()
    End
End Sub
Sub mnuHelpAbout_Click ()
    frmAbout.Show Modal
End Sub
Sub mnuLibrarianFacets_Click ()
    frmFacets.Show Modal
End Sub
Sub mnuLibrarianVocabulary_Click ()
    frmVocabulary.Show Modal
End Sub
Sub mnuReportsArtifact_Click ()
    frmArtifactUsage.Show Modal
End Sub
Sub mnuReportsUser_Click ()
    frmUserUsage.Show Modal
End Sub
Sub mnuRole_Click ()
    frmRole.Show Modal
End Sub
Sub mnuThesaurus_Click ()
    frmThesaurus.Show Modal
End Sub
Sub mnuUser_Click ()
    frmUser.Show Modal
End Sub
Sub mnuUserInsert_Click ()
    frmDetails.Show Modal
End Sub
'------------------------------------------------------------------------
'Name: mnuUserSearch_Click
'
'Type: Subroutine
```

```
'
'Parameters: None
'
'Returns: None
'
'Description: Search command selected.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
'
Sub mnuUserSearch_Click ()

    frmSearch.Show Modal

End Sub

Sub picStatusbar_Resize ()

    pnl3dStatusbar.Left = 0
    pnl3dStatusbar.Top = 0
    pnl3dStatusbar.Height = picStatusbar.Height
    pnl3dStatusbar.Width = picStatusbar.Width

End Sub

Sub picToolbar_Resize ()

    'Whenever the picture box is resized, resize the panel also.
    pnl3dToolbar.Left = 0
    pnl3dToolbar.Top = 0
    pnl3dToolbar.Height = picToolbar.Height
    pnl3dToolbar.Width = picToolbar.Width

End Sub

Sub pnl3dToolbar_MouseMove (Button As Integer, Shift As Integer, X As Single, Y As Single)

    pnl3dStatusbar.Caption = ""

End Sub

'frmRole
VERSION 2.00
Begin Form frmRole
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Role"
   ClientHeight    =   5175
   ClientLeft      =   240
   ClientTop       =   1590
   ClientWidth     =   8295
   Height          =   5580
   Left            =   180
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   ScaleHeight     =   5175
   ScaleWidth      =   8295
   Top             =   1245
   Width           =   8415
   Begin SSPanel Panel3D1
      BevelWidth      =   3
      BorderWidth     =   1
      Height          =   5175
      Left            =   0
      TabIndex        =   8
      Top             =   0
      Width           =   8295
```

```
Begin SSFrame Frame3D2
   Caption          =   "&Commands"
   ForeColor        =   &H00000000&
   Height           =   3735
   Left             =   6360
   ShadowStyle      =   1  'Raised
   TabIndex         =   15
   Top              =   240
   Width            =   1695
   Begin SSCommand cmd3dAdd
      AutoSize         =   2  'Adjust Button Size To Picture
      Caption          =   "&Add"
      Height           =   975
      Left             =   360
      Picture          =   RWFROLE.FRX:0000
      TabIndex         =   5
      Top              =   480
      Width            =   975
   End
   Begin SSCommand cmd3dRemove
      AutoSize         =   2  'Adjust Button Size To Picture
      Caption          =   "&Remove"
      Height           =   975
      Left             =   360
      Picture          =   RWFROLE.FRX:0302
      TabIndex         =   6
      Top              =   1440
      Width            =   975
   End
   Begin SSCommand cmd3dDone
      AutoSize         =   2  'Adjust Button Size To Picture
      Caption          =   "&Done"
      Height           =   975
      Left             =   360
      Picture          =   RWFROLE.FRX:0604
      TabIndex         =   7
      Top              =   2400
      Width            =   975
   End
End
Begin SSFrame Frame3D1
   Caption          =   "Role"
   ForeColor        =   &H00000000&
   Height           =   4695
   Left             =   240
   ShadowStyle      =   1  'Raised
   TabIndex         =   9
   Top              =   240
   Width            =   5655
   Begin SSPanel Panel3D3
      AutoSize         =   3  'AutoSize Child To Panel
      BevelOuter       =   1  'Inset
      BevelWidth       =   3
      BorderWidth      =   1
      Height           =   1380
      Left             =   720
      TabIndex         =   12
      Top              =   3000
      Width            =   4695
      Begin ListBox lstNameComments
         FontBold         =   -1  'True
         FontItalic       =   0   'False
         FontName         =   "Courier New"
         FontSize         =   8.25
         FontStrikethru   =   0   'False
         FontUnderline    =   0   'False
         Height           =   1290
```

```
      Left            =   45
      MultiSelect     =   1  'Simple
      TabIndex        =   4
      Top             =   45
      Width           =   4605
   End
End
Begin SSOption opt3dRemoveRole
   Caption         =   "&Remove Role"
   Height          =   375
   Left            =   480
   TabIndex        =   3
   TabStop         =   0   'False
   Top             =   2520
   Width           =   1815
End
Begin SSPanel Panel3D2
   AutoSize        =   3  'AutoSize Child To Panel
   BevelOuter      =   1  'Inset
   BevelWidth      =   3
   BorderWidth     =   1
   Height          =   375
   Left            =   720
   TabIndex        =   11
   Top             =   960
   Width           =   2055
   Begin TextBox txtName
      Height          =   285
      Left            =   45
      TabIndex        =   1
      Top             =   45
      Width           =   1965
   End
End
Begin SSOption opt3dAddRole
   Caption         =   "&Add Role"
   Height          =   255
   Left            =   480
   TabIndex        =   0
   Top             =   360
   Width           =   1215
End
Begin SSPanel Panel3D4
   AutoSize        =   3  'AutoSize Child To Panel
   BevelOuter      =   1  'Inset
   BevelWidth      =   3
   BorderWidth     =   1
   Height          =   1335
   Left            =   3000
   TabIndex        =   10
   Top             =   960
   Width           =   2415
   Begin TextBox txtComments
      Height          =   1245
      Left            =   45
      MultiLine       =   -1  'True
      TabIndex        =   2
      Top             =   45
      Width           =   2325
   End
End
Begin Label Label1
   BackColor       =   &H00C0C0C0&
   Caption         =   "Name:"
   Height          =   255
   Left            =   720
   TabIndex        =   14
```

```
            Top             =   720
            Width           =   855
         End
         Begin Label Label2
            BackColor       =   &H00C0C0C0&
            Caption         =   "Comments:"
            Height          =   255
            Left            =   3120
            TabIndex        =   13
            Top             =   720
            Width           =   1095
         End
      End
   End
End
Option Explicit

'------------------------------------------------------------------------
'Name: AddRole
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This routine adds a role to the database.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub AddRole ()

    '--------------------
    'Variable Declaration
    '--------------------
    'Role information.
    Dim sRoleName As String
    Dim sRoleComments As String

    'Qurey to obtain the role ids.
    Dim sqryRoleId As String
    Dim sRoleId As String
    Dim nMaxRoleId As Integer
    Dim nRoleId As Integer

    'Dynaset containing the Role ids.
    Dim dsRoleId As Dynaset

    'Message box variables.
    Dim sMessage As String
    Dim sTitle As String
    Dim nResponse As Integer

    Dim dsUser As Dynaset
    Dim tblRole As table

    '--------------
    'Check Text Box
    '--------------
    If (txtName.Text = "") Then
        sMessage = "Name box is empty. Null user can not be added."
        sTitle = "Add Role"
        nResponse = MsgBox(sMessage, MB_OK + MB_ICONEXCLAMATION, sTitle)
        Exit Sub
    End If
```

```
    '-------------------------------------
    'Role and the description to be added
    '-------------------------------------
    sRoleName = txtName.Text
    sRoleComments = txtComments.Text


    '------------------
    'Get the maximum-id
    '------------------
    sqryRoleId = "SELECT RoleId FROM Role"
    Set dsRoleId = gdbRwf.CreateDynaset(sqryRoleId)

    Do Until dsRoleId.EOF
        sRoleId = dsRoleId("RoleId")
        If nMaxRoleId < CInt(sRoleId) Then
            nMaxRoleId = CInt(sRoleId)
        End If
        dsRoleId.MoveNext
    Loop

    nRoleId = nMaxRoleId + 1


    '--------------
    'Add New Record
    '--------------
    Set tblRole = gdbRwf.OpenTable("Role")

    tblRole.AddNew
    tblRole("RoleName") = sRoleName
    tblRole("RoleComments") = sRoleComments
    tblRole("RoleId") = nRoleId
    tblRole.Update
    tblRole.Close

End Sub

Sub cmd3dAdd_Click ()


    '--------------------
    'Add User to Database
    '--------------------
    AddRole

    '--------------------
    'Update all the boxes
    '--------------------
    FillBoxes

    '---------------------
    'Set focus to Name Box
    '---------------------
    txtName.SetFocus


End Sub

Sub cmd3dDone_Click ()
    '---------------------
    'Variable Declarations
    '---------------------
    'Message box variables.
    Dim sMessage As String
```

```
    Dim sTitle As String
    Dim nResponse As Integer

    'Check the Text Box to see the user is mistakenly closing.
    If txtName.Text <> "" Then

        sMessage = "Name box is not empty, may be you are trying to add a Role."
        sMessage = sMessage + "  Do you want to go back?"
        sTitle = "Add Role Message"

        nResponse = MsgBox(sMessage, MB_YESNO, sTitle)

        'if user wants to go back, exit this subroutine.
        If nResponse = IDYES Then
            'Give focus to text box.
            EnableAddRole
            txtName.SetFocus

            Exit Sub
        End If

      End If


    Unload Me

End Sub

Sub cmd3dRemove_Click ()

    RemoveRole


End Sub

'------------------------------------------------------------------------
'Name: EnableAddRole
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function enables all the add Role boxes and buttons,
'where as the EnableRemoveRole does the vice versa.
'------------------------------------------------------------------------
'
Sub EnableAddRole ()

    '------
    'Enable
    '------
    txtName.Enabled = True
    txtComments.Enabled = True
    cmd3dAdd.Enabled = True

    '-------
    'Disable
    '-------
    opt3dRemoveRole.Value = False
    lstNameComments.Enabled = False
    cmd3dRemove.Enabled = False

End Sub
```

```
'------------------------------------------------------------------------
'Name: EnableRemoveRole
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function enables all the remove Role boxes and buttons,
'where as the EnableAddRole does the vice versa.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub EnableRemoveRole ()

    '------
    'Enable
    '------
    lstNameComments.Enabled = True
    cmd3dRemove.Enabled = True

    '-------
    'Disable
    '-------
    opt3dAddRole.Value = False
    txtName.Enabled = False
    txtComments.Enabled = False
    cmd3dAdd.Enabled = False

End Sub

'------------------------------------------------------------------------
'Name: FillBoxes
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This subroutine fills all the boxes, i.e. text box, listbox
'with information. This can also be called after ADD or REMOVE is done.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub FillBoxes ()

    Dim sqryName As String
    Dim sNameDescription As String
    Dim dsName As Dynaset


    '------------------
    'Get Name and Group
    '------------------
    sqryName = "SELECT RoleName, RoleComments FROM Role"
    Set dsName = gdbRwf.CreateDynaset(sqryName)

    '--------------
    'Empty Name Box
    '--------------
    txtName.Text = ""
    txtComments.Text = ""
```

```
    '------------------------------
    'Fill Name+Description List Box
    '------------------------------
    'First clear the list box.
    lstNameComments.Clear

    Do Until dsName.EOF
        sNameDescription = "(" + dsName("RoleName") + ")"
        sNameDescription = sNameDescription + Space$(20 - Len(sNameDescription)) + " : "
        sNameDescription = sNameDescription + "(" + dsName("RoleComments") + ")"
        lstNameComments.AddItem sNameDescription
        dsName.MoveNext
    Loop

End Sub

Sub Form_Load ()

    '---------------
    'Center the form
    '---------------
    CenterForm Me

    '-----------------------------------
    'Fill boxes on the form with details
    '-----------------------------------
    FillBoxes

End Sub

Sub opt3dAddRole_Click (Value As Integer)

    EnableAddRole

    '---------------------
    'Set focus to Name Box
    '---------------------
    txtName.SetFocus


End Sub

Sub opt3dRemoveRole_Click (Value As Integer)

    EnableRemoveRole

End Sub

'-------------------------------------------------------------------------
'Name: RemoveRole
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function removes one or more users selected in the
'remove list box.
'
'Author: Ram Mareddy
'-------------------------------------------------------------------------
'
Sub RemoveRole ()

    '---------------------
```

```
    'Variable Declarations
    '--------------------
    'Loop counter.
    Dim i As Integer

    'Full string in the remove box.
    Dim sNameComments As String

    'Name of the Role.
    Dim sName As String

    'Start and end for string parsing.
    Dim nStart As Integer
    Dim nLength As Integer

    'Query to remove the Role.
    Dim sqryRemoveRole As String


    '---------------------------------------
    'All the items selected should be removed
    '---------------------------------------
    For i = 0 To lstNameComments.ListCount - 1
        If lstNameComments.Selected(i) = True Then
            sNameComments = lstNameComments.List(i)

            'Parse the sNameComments into name and comments

            'Format of name is (name)  : (comments), length of name is position 2 to
            'the character before ).
            nStart = 2
            nLength = InStr(1, sNameComments, ")") - nStart
            sName = Mid$(sNameComments, nStart, nLength)

            sqryRemoveRole = "DELETE FROM Role WHERE RoleName = '" + sName + "'"
            gdbRwf.Execute (sqryRemoveRole)
        End If
    Next


    '-------------------
    'Refresh boxes again
    '-------------------
    FillBoxes

End Sub

'frmRetrieval
VERSION 2.00
Begin Form frmRetrieval
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Retrieval"
   ClientHeight    =   4095
   ClientLeft      =   270
   ClientTop       =   1545
   ClientWidth     =   7320
   Height          =   4500
   Left            =   210
   LinkTopic       =   "Form2"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   ScaleHeight     =   4095
   ScaleWidth      =   7320
   Top             =   1200
   Width           =   7440
   Begin SSPanel panDate
      BevelWidth      =   2
```

```
      Height            =   4095
      Left              =   0
      TabIndex          =   0
      Top               =   0
      Width             =   7335
      Begin SSFrame Frame3D1
         Caption        =   "&Artefact Description"
         ForeColor      =   &H00000000&
         Height         =   3615
         Left           =   240
         ShadowStyle    =   1  'Raised
         TabIndex       =   1
         Top            =   240
         Width          =   5055
         Begin SpreadSheet sprdRetrieval
            AllowResize     =   -1  'True
            FontBold        =   -1  'True
            FontItalic      =   0   'False
            FontName        =   "MS Sans Serif"
            FontSize        =   8.25
            FontStrikethru  =   0   'False
            FontUnderline   =   0   'False
            Height          =   3015
            InterfaceDesigner=   RWFRTRV.FRX:0000
            Left            =   240
            MaxCols         =   6
            MaxRows         =   20
            TabIndex        =   2
            Top             =   360
            Width           =   4575
         End
      End
      Begin SSFrame Frame3D3
         Caption        =   "&Commands"
         ForeColor      =   &H00000000&
         Height         =   2655
         Left           =   5520
         ShadowStyle    =   1  'Raised
         TabIndex       =   3
         Top            =   240
         Width          =   1575
         Begin SSCommand cmdReuse
            AutoSize        =   2  'Adjust Button Size To Picture
            Caption         =   "&Reuse"
            Height          =   975
            Left            =   360
            Picture         =   RWFRTRV.FRX:0004
            TabIndex        =   5
            Top             =   360
            Width           =   975
         End
         Begin SSCommand cmd3dDone
            AutoSize        =   2  'Adjust Button Size To Picture
            Caption         =   "&Done"
            Height          =   975
            Left            =   360
            Picture         =   RWFRTRV.FRX:0306
            TabIndex        =   4
            Top             =   1320
            Width           =   975
         End
      End
   End
End
Option Explicit
Const HEADING_ROW_NUMBER = 0
Const FILE_NAME_COLUMN_NUMBER = 0
```

```
Const PATH_COLUMN_NUMBER = 1
Const SIZE_COLUMN_NUMBER = 2
Const DESCRIPTION_COLUMN_NUMBER = 3
Const AUTHOR_COLUMN_NUMBER = 4
Const DATE_OF_INSERTION_COLUMN_NUMBER = 5

Sub cmd3dDone_Click ()

    Unload Me

End Sub

Sub cmdReuse_Click ()

    '------------
    'Declarations
    '------------
    'Queries
    Dim sqryUserId As String
    Dim sqryArtifactId As String
    Dim sqryEventId As String

    'Dynasets
    Dim dsUserId As Dynaset
    Dim dsArtifactId As Dynaset
    Dim dsEventId As Dynaset
    Dim tblEvent As Table

    'Ids
    Dim nUserId As String
    Dim sArtifactFileName As String
    Dim sArtifactPath As String
    Dim nArtifactId As Integer
    Dim nEventId As Integer
    Dim nMaxEventId As Integer

    'Record the reuse event.
    'Get the user id.
    sqryUserId = "SELECT UserId FROM USERS WHERE UserFirstName = '" + gsLoginFirstName +
"'" + " AND UserLastName = '" + gsLoginLastName + "'"
    Set dsUserId = gdbRwf.CreateDynaset(sqryUserId)

    nUserId = dsUserId("UserId")

    'Get the artifact id.
    sprdRetrieval.Row = sprdRetrieval.ActiveRow
    sprdRetrieval.Col = 0
    sArtifactFileName = sprdRetrieval.Text

    sprdRetrieval.Col = 1
    sArtifactPath = sprdRetrieval.Text

    sqryArtifactId = "SELECT ArtifactId FROM ARTIFACT WHERE ArtifactFileName = '" +
sArtifactFileName + "'" + " AND ArtifactPath = '" + sArtifactPath + "'"
    Set dsArtifactId = gdbRwf.CreateDynaset(sqryArtifactId)

    nArtifactId = dsArtifactId("ArtifactId")

    'Get the maximum event id.
    sqryEventId = "SELECT EventId FROM EVENT"
    Set dsEventId = gdbRwf.CreateDynaset(sqryEventId)

    nMaxEventId = 1
    Do Until dsEventId.EOF
        nEventId = CInt(dsEventId("EventId"))
        If nMaxEventId < nEventId Then
            nMaxEventId = nEventId
```

```
        End If
        dsEventId.MoveNext
    Loop
    nEventId = nMaxEventId + 1

    'Login the event
    Set tblEvent = gdbRwf.OpenTable("Event")

    tblEvent.AddNew
    tblEvent("EventId") = nEventId
    tblEvent("ArtifactId") = nArtifactId
    tblEvent("UserId") = nUserId
    tblEvent.Update
    tblEvent.Close

End Sub

'---------------------------------------------------------------------------
'Name: FillSpread
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: Spreadsheet in the current form is filled with the requested
'information.
'
'Author: Ram Mareddy
'---------------------------------------------------------------------------
'
Sub FillSpread ()

    If frmSearch.sprdSearch.MaxCols = 1 Then
        RetrieveWithOutSynonyms gsClassification
    Else
        RetrieveWithSynonyms
    End If


End Sub

Sub Form_Load ()

    '---------------
    'Center the form
    '---------------
    CenterForm Me


    '----------
    'Set spread
    '----------
    SetSpread


    '-----------
    'Fill Spread
    '-----------
    FillSpread


End Sub

'---------------------------------------------------------------------------
```

```
'Name: RetrieveWithOutSynonyms
'
'Type: Subroutine
'
'Parameters: sClassification
'
'Returns: None
'
'Description: Spreadsheet on the Retrieval form is filled with the details of
'artifacts matching the classification.
'
'Author: Ram Mareddy
'-------------------------------------------------------------------------
'
Sub RetrieveWithOutSynonyms (sClassification As String)

    '---------------------
    'Variable Declarations
    '---------------------
    'String that holds the query for the vocabulary id.
    Dim sqryClassification As String

    'Dynaset that holds the results.
    Dim dsClassification As Dynaset

    'Row number for the current artifact.
    Dim nCurrentRow As Integer

    'Classification string of the current artifact.
    Dim sCurrentArtifact As String


    '------------------------
    'Query for classification
    '------------------------
    sqryClassification = "SELECT VocabularyIdString, ArtifactFileName, ArtifactPath,
ArtifactSize, ArtifactDescription, ArtifactAuthor, artifactDateOfInsertion FROM ARTIFACT"
    Set dsClassification = gdbRwf.CreateDynaset(sqryClassification)

    nCurrentRow = 1
    Do Until dsClassification.EOF
        sCurrentArtifact = dsClassification("VocabularyIdString")
        If IsArtifactInThisClass(sClassification, sCurrentArtifact) = True Then
            sprdRetrieval.Row = nCurrentRow

            sprdRetrieval.Col = FILE_NAME_COLUMN_NUMBER
            sprdRetrieval.Text = dsClassification("ArtifactFileName")

            sprdRetrieval.Col = PATH_COLUMN_NUMBER
            sprdRetrieval.Text = dsClassification("ArtifactPath")

            sprdRetrieval.Col = SIZE_COLUMN_NUMBER
            sprdRetrieval.Text = dsClassification("ArtifactSize")

            sprdRetrieval.Col = DESCRIPTION_COLUMN_NUMBER
            sprdRetrieval.Text = dsClassification("ArtifactDescription")

            sprdRetrieval.Col = AUTHOR_COLUMN_NUMBER
            sprdRetrieval.Text = dsClassification("ArtifactAuthor")

            sprdRetrieval.Col = DATE_OF_INSERTION_COLUMN_NUMBER
            sprdRetrieval.Text = dsClassification("ArtifactDateOfInsertion")

            nCurrentRow = nCurrentRow + 1
        End If

        dsClassification.MoveNext
```

```
    Loop

    sprdRetrieval.MaxRows = nCurrentRow - 1

End Sub

'--------------------------------------------------------------------------
'Name: RetrieveWithSynonyms
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: Spreadsheet on the Retrieval form is filled with the details of
'artifacts obtained from synonym array.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
'
Sub RetrieveWithSynonyms ()

    '--------------------
    'Variable Declarations
    '--------------------
    'String that holds the query for the vocabulary id.
    Dim sqryClassification As String

    'Dynaset that holds the results.
    Dim dsClassification As Dynaset

    'Row number for the current artifact.
    Dim nCurrentRow As Integer

    'Classification string of the current artifact.
    Dim sCurrentArtifact As String
    Dim nCounter As Integer


    '------------------------
    'Query for classification
    '------------------------
    'Get all the synonyms in the global array.
    For nCounter = 1 To UBound(gsSynonymHits)


        sqryClassification = "SELECT ArtifactFileName, ArtifactPath, ArtifactSize,
ArtifactDescription, ArtifactAuthor, artifactDateOfInsertion FROM ARTIFACT where
VocabularyIdString = '" + gsSynonymHits(nCounter) + "'"
        Set dsClassification = gdbRwf.CreateDynaset(sqryClassification)

        nCurrentRow = 1
        sprdRetrieval.Row = nCurrentRow

        sprdRetrieval.Col = FILE_NAME_COLUMN_NUMBER
        sprdRetrieval.Text = dsClassification("ArtifactFileName")

        sprdRetrieval.Col = PATH_COLUMN_NUMBER
        sprdRetrieval.Text = dsClassification("ArtifactPath")

        sprdRetrieval.Col = SIZE_COLUMN_NUMBER
        sprdRetrieval.Text = dsClassification("ArtifactSize")

        sprdRetrieval.Col = DESCRIPTION_COLUMN_NUMBER
        sprdRetrieval.Text = dsClassification("ArtifactDescription")
```

```
        sprdRetrieval.Col = AUTHOR_COLUMN_NUMBER
        sprdRetrieval.Text = dsClassification("ArtifactAuthor")

        sprdRetrieval.Col = DATE_OF_INSERTION_COLUMN_NUMBER
        sprdRetrieval.Text = dsClassification("ArtifactDateOfInsertion")

        nCurrentRow = nCurrentRow + 1

        dsClassification.MoveNext
    Next

    sprdRetrieval.MaxRows = nCurrentRow - 1


End Sub

'--------------------------------------------------------------------------------
'Name: SetSpread
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function sets up the spread on the main form.
'
'Author : Ram Mareddy
'--------------------------------------------------------------------------------
'
Sub SetSpread ()


    '------------
    'Declarations
    '------------

    'Row heights and column widths.
    Const HEADING_ROW_HEIGHT = 20
    Const FILE_NAME_COLUMN_WIDTH = 17.9
    Const PATH_COLUMN_WIDTH = 30
    Const SIZE_COLUMN_WIDTH = 20
    Const DESCRIPTION_COLUMN_WIDTH = 20
    Const AUTHOR_COLUMN_WIDTH = 20
    Const DATE_OF_INSERTION_COLUMN_WIDTH = 20



    '----------------------------
    'Row Heights and column widths
    '----------------------------
    'Heading row.
    sprdRetrieval.RowHeight(HEADING_ROW_NUMBER) = HEADING_ROW_HEIGHT

    'Column widths
    sprdRetrieval.ColWidth(FILE_NAME_COLUMN_NUMBER) = FILE_NAME_COLUMN_WIDTH
    sprdRetrieval.ColWidth(PATH_COLUMN_NUMBER) = PATH_COLUMN_WIDTH
    sprdRetrieval.ColWidth(SIZE_COLUMN_NUMBER) = SIZE_COLUMN_WIDTH
    sprdRetrieval.ColWidth(DESCRIPTION_COLUMN_NUMBER) = DESCRIPTION_COLUMN_WIDTH
    sprdRetrieval.ColWidth(AUTHOR_COLUMN_NUMBER) = AUTHOR_COLUMN_WIDTH
    sprdRetrieval.ColWidth(DATE_OF_INSERTION_COLUMN_NUMBER) =
DATE_OF_INSERTION_COLUMN_WIDTH


    '-------------
    'Column Titles
    '-------------
```

```
    sprdRetrieval.FontSize = gudtRwfIni.nFontSize
    sprdRetrieval.FontName = gudtRwfIni.sFontName
    sprdRetrieval.FontBold = gudtRwfIni.bFontBold

    SpreadSetText sprdRetrieval, FILE_NAME_COLUMN_NUMBER, HEADING_ROW_NUMBER, "File"
    SpreadSetText sprdRetrieval, PATH_COLUMN_NUMBER, HEADING_ROW_NUMBER, "Path"
    SpreadSetText sprdRetrieval, SIZE_COLUMN_NUMBER, HEADING_ROW_NUMBER, "Size"
    SpreadSetText sprdRetrieval, DESCRIPTION_COLUMN_NUMBER, HEADING_ROW_NUMBER,
"Description"
    SpreadSetText sprdRetrieval, AUTHOR_COLUMN_NUMBER, HEADING_ROW_NUMBER, "Author"
    SpreadSetText sprdRetrieval, DATE_OF_INSERTION_COLUMN_NUMBER, HEADING_ROW_NUMBER,
"Date Of Insertion"


End Sub

'frmSearch
VERSION 2.00
Begin Form frmSearch
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Search"
   ClientHeight    =   4095
   ClientLeft      =   2550
   ClientTop       =   2190
   ClientWidth     =   7320
   Height          =   4500
   Left            =   2490
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   ScaleHeight     =   4095
   ScaleWidth      =   7320
   Top             =   1845
   Width           =   7440
   Begin SSPanel panDate
      BevelWidth      =   2
      Height          =   4095
      Left            =   0
      TabIndex        =   0
      Top             =   0
      Width           =   7335
      Begin SSFrame Frame3D3
         Caption         =   "&Commands"
         ForeColor       =   &H00000000&
         Height          =   2535
         Left            =   5520
         ShadowStyle     =   1  'Raised
         TabIndex        =   3
         Top             =   240
         Width           =   1575
         Begin SSCommand cmd3dSearch
            AutoSize        =   2  'Adjust Button Size To Picture
            Caption         =   "&Search"
            Height          =   975
            Left            =   240
            Picture         =   RWFSRCH.FRX:0000
            TabIndex        =   5
            Top             =   360
            Width           =   1095
         End
         Begin SSCommand cmd3dCancel
            AutoSize        =   2  'Adjust Button Size To Picture
            Caption         =   "&Cancel"
            Height          =   975
            Left            =   240
            Picture         =   RWFSRCH.FRX:0302
            TabIndex        =   4
```

```
            Top             =   1320
            Width           =   1095
         End
      End
      Begin SSFrame Frame3D1
         Caption         =   "&Artefact Description"
         ForeColor       =   &H00000000&
         Height          =   3615
         Left            =   240
         ShadowStyle     =   1  'Raised
         TabIndex        =   1
         Top             =   240
         Width           =   5055
         Begin SpreadSheet sprdSearch
            AllowResize     =   -1  'True
            FontBold        =   -1  'True
            FontItalic      =   0   'False
            FontName        =   "MS Sans Serif"
            FontSize        =   8.25
            FontStrikethru  =   0   'False
            FontUnderline   =   0   'False
            Height          =   3015
            InterfaceDesigner=   RWFSRCH.FRX:0604
            Left            =   240
            MaxCols         =   1
            MaxRows         =   20
            TabIndex        =   2
            Top             =   360
            Width           =   4575
         End
      End
   End
End
Option Explicit

'Row and column numbers.
Const HEADING_ROW_NUMBER = 0
Const FACET_COLUMN_NUMBER = 0
Const VOCABULARY_COLUMN_NUMBER = 1
Const SYNONYM_COLUMN_NUMBER = 2

Sub cmd3dCancel_Click ()

    Unload Me

End Sub

Sub cmd3dSearch_Click ()

    '------------------------------
    'Search with or without synonyms
    '------------------------------
    If sprdSearch.MaxCols = 1 Then
        SearchWithOutSynonyms

    Else
        SearchWithSynonyms
    End If


End Sub

'--------------------------------------------------------------------------------
'Name: FillSpread
'
'Type: Subroutine
'
```

```
'Parameters: None
'
'Returns: None
'
'Description:
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------------
'
Sub FillSpread ()

    '--------------------
    'Variable Declarations
    '--------------------
    'Number of facets in the system.
    Dim nNumFacets As Integer

    'Loop counters.
    Dim i As Integer
    Dim j As Integer

    'Number of columns in the spread.
    Const NUM_COLUMNS_IN_SPREAD = 2


    '-------------------------------------
    'Get the number of facets in the system
    '-------------------------------------
    nNumFacets = GetNumFacets()


    '-----------------------------------------
    'Set the maximum number of rows in the spread
    '-----------------------------------------
    sprdSearch.MaxRows = nNumFacets


    '---------------
    'Clear the Spread
    '---------------
    For i = 0 To NUM_COLUMNS_IN_SPREAD
        For j = 1 To nNumFacets
            sprdSearch.Row = j
            sprdSearch.Col = i
            sprdSearch.Text = ""
        Next
    Next


    '--------------------
    'Fill facets in search
    '--------------------
    FillFacetsInSearch


    '-------------------------
    'Fill Vocabulary in search
    '-------------------------
    'Initially put ALL in all the vocabulary.
    For j = 1 To nNumFacets
        sprdSearch.Row = j
        sprdSearch.Col = VOCABULARY_COLUMN_NUMBER
        sprdSearch.Text = "ALL"
    Next j


End Sub
```

```
Sub Form_Load ()

    '---------------
    'Center the form
    '---------------
    CenterForm Me


    '----------
    'Set spread
    '----------
    SetSpread


    '-----------
    'Fill Spread
    '-----------
    FillSpread


End Sub

'--------------------------------------------------------------------------------
'Name: GetClassificationString
'
'Type: Subroutine
'
'Parameters: sClassification
'
'Returns: None
'
'Description: Gets the classification for the currently requested artifacts.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------------
'
Sub GetClassificationString (sClassification As String)

    '--------------------
    'Variable Declaration
    '--------------------
    'Loop counter.
    Dim i As Integer

    'Selected vocabulary for the facet.
    Dim sVocabularyName As String

    'Id of the vocabulary for the facet.
    Dim nVocabularyId As Integer


    '-------------------------------------------------------
    'Get Vocabulary Ids and classification for current selection
    '-------------------------------------------------------
    For i = 1 To sprdSearch.MaxRows
        'Get vocabulary name.
        sprdSearch.Col = VOCABULARY_COLUMN_NUMBER
        sprdSearch.Row = i
        sVocabularyName = sprdSearch.Text

        'Get vocabulary id.
        nVocabularyId = GetVocabularyId(sVocabularyName)

        'Get the classification.
        If i <> 1 Then
```

```
            sClassification = sClassification + "," + CStr(nVocabularyId)
        Else
            sClassification = CStr(nVocabularyId)
        End If

    Next i

End Sub

'-------------------------------------------------------------------------
'Name: GetSynonymFlagString
'
'Type: Subroutine
'
'Parameters: sSynonymFlagString
'
'Returns: None
'
'Description: The facets that are okd for the use of synonyms are returned
'in the form a string.
'
'Author: Ram Mareddy
'-------------------------------------------------------------------------
'
Sub GetSynonymFlagString (sSynonymFlagString As String)

    '--------------------
    'Variable Declaration
    '--------------------
    'Loop counter.
    Dim i As Integer

    'Synonym flags
    Dim sSynonymFlag As String
    Dim nSynonymFlag As Integer

    '-------------------------------------------------------------
    'Get Vocabulary Ids and classification for current selection
    '-------------------------------------------------------------
    For i = 1 To sprdSearch.MaxRows
        'Get vocabulary name.
        sprdSearch.Col = SYNONYM_COLUMN_NUMBER
        sprdSearch.Row = i
        sSynonymFlag = sprdSearch.Text

        If sSynonymFlag = "YES" Then
            nSynonymFlag = 1
        Else
            nSynonymFlag = 0
        End If

        'Make the classification.
        If i <> 1 Then
            sSynonymFlagString = sSynonymFlagString + "," + CStr(nSynonymFlag)
        Else
            sSynonymFlagString = CStr(nSynonymFlag)
        End If

    Next i


End Sub

Sub PresentArtifacts ()

    frmRetrieval.Show Modal
```

```
End Sub

'------------------------------------------------------------------------------
'Name: ResetSpread
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: The appearance of spread is reset.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------------
'
Sub ResetSpread ()

    '------------
    'Declarations
    '------------
    'Loop counter.
    Dim i As Integer

    'Row heights and column widths.
    Const HEADING_ROW_HEIGHT = 20
    Const FACET_COLUMN_WIDTH = 14
    Const VOCABULARY_COLUMN_WIDTH = 12
    Const SYNONYM_COLUMN_WIDTH = 9.5


    '-----------------------------
    'Increase the number of columns
    '-----------------------------
    sprdSearch.MaxCols = 2


    '-----------------------------
    'Row Heights and column widths
    '-----------------------------
    'Heading row.
    sprdSearch.RowHeight(HEADING_ROW_NUMBER) = HEADING_ROW_HEIGHT

    'Column widths
    sprdSearch.ColWidth(FACET_COLUMN_NUMBER) = FACET_COLUMN_WIDTH
    sprdSearch.ColWidth(VOCABULARY_COLUMN_NUMBER) = VOCABULARY_COLUMN_WIDTH
    sprdSearch.ColWidth(SYNONYM_COLUMN_NUMBER) = SYNONYM_COLUMN_WIDTH


    '-------------
    'Column Titles
    '-------------
    SpreadSetText sprdSearch, SYNONYM_COLUMN_NUMBER, HEADING_ROW_NUMBER, "Synonyms"


    '------------------
    'Column Appearances
    '------------------
    sprdSearch.Row = SS_ALL_ROWS
    sprdSearch.Col = SYNONYM_COLUMN_NUMBER
    sprdSearch.TypeHAlign = SS_CELL_H_ALIGN_RIGHT


    '-------------------------------
    'Fill the synonym column with NOs
    '-------------------------------
```

```
    For i = 1 To sprdSearch.MaxRows
        sprdSearch.Row = i
        sprdSearch.Col = SYNONYM_COLUMN_NUMBER
        sprdSearch.Text = "NO"
    Next i

End Sub

'--------------------------------------------------------------------------
'Name: SearchWithOutSynonyms
'
'Type: Subroutine
'
'Parameters:
'
'Returns: None
'
'Description: Matching artifacts for the given classification is searched.
'In this case, no synonyms are used.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
'
Sub SearchWithOutSynonyms ()

    '--------------------
    'Variable Declaration
    '--------------------
    'Classification string.
    Dim sClassification As String

    'Loop counter.
    Dim i As Integer

    'Artifacts fitting the classificaion.
    Dim nNumHits As Integer

    'Message box variables.
    Dim sMessage As String
    Dim sTitle As String
    Dim nResponse As Integer


    '------------------------------------------------------------
    'Get Vocabulary Ids and classification for current selection
    '------------------------------------------------------------
    GetClassificationString sClassification


    '---------------------
    'Search the repository
    '---------------------
    nNumHits = SearchRepWithOutSynonyms(sClassification)
    If nNumHits = 0 Then
        sMessage = "There are no artifacts present in the system with your
 classification."
        sMessage = sMessage + " Do you want to use synonyms?"
        sTitle = "Search the Repository"
        nResponse = MsgBox(sMessage, MB_YESNO + MB_ICONSTOP, sTitle)

        'If answer is YES, then we need to search with synonyms.
        If nResponse = IDYES Then
            SynonymPreparation
        End If

    Else
        If nNumHits > 1 Then
```

```
            sMessage = "There are " + CStr(nNumHits) + " artifacts present in the system
with your classification."
            sMessage = sMessage + " Do you want to view them?"
        Else
            sMessage = "There is 1 artifact present in the system with your
classification."
            sMessage = sMessage + " Do you want to view that?"
        End If
        sTitle = "Search the Repository"
        nResponse = MsgBox(sMessage, MB_YESNO, sTitle)

        If nResponse = IDYES Then
            'If answer is YES, then we need to present the artifacts.
            gsClassification = sClassification
            PresentArtifacts
        End If

    End If


End Sub

'--------------------------------------------------------------------------
'Name: SearchWithSynonyms
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: Matching artifacts for the given classification is searched.
'In this case, synonyms are used.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
'
Sub SearchWithSynonyms ()

    '--------------------
    'Variable Declaration
    '--------------------
    'Classification string.
    Dim sClassification As String

    'Synonym string.
    Dim sSynonymFlagString As String

    'Artifacts fitting the classificaion.
    Dim nNumHits As Integer

    'Message box variables.
    Dim sMessage As String
    Dim sTitle As String
    Dim nResponse As Integer


    '-----------------------------------------------------
    'Get Vocabulary Ids and classification for current selection
    '-----------------------------------------------------
    GetClassificationString sClassification


    '-----------------------------
    'Get the YES or NO for synonyms
    '-----------------------------
    GetSynonymFlagString sSynonymFlagString
```

```
    '--------------------
    'Search the repository
    '--------------------
    nNumHits = SearchRepWithSynonyms(sClassification, sSynonymFlagString)
    If nNumHits = 0 Then
        sMessage = "There are no artifacts present in the system with your classification

        sMessage = sMessage + " and the synonyms you selected. Try more generalizing."
        sTitle = "Search the Repository"
        nResponse = MsgBox(sMessage, MB_OK + MB_ICONSTOP, sTitle)

    Else
        If nNumHits > 1 Then
            sMessage = "There are " + CStr(nNumHits) + " artifacts present in the system
with your classification."
            sMessage = sMessage + " Do you want to view them?"
        Else
            sMessage = "There is 1 artifact present in the system with your
classification."
            sMessage = sMessage + " Do you want to view that?"
        End If
        sTitle = "Search the Repository"
        nResponse = MsgBox(sMessage, MB_YESNO, sTitle)

        If nResponse = IDYES Then
            'If answer is YES, then we need to present the artifacts.
            gsClassification = sClassification
            PresentArtifacts
        End If

    End If


End Sub

'-------------------------------------------------------------------------------
'Name: SetSpread
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function sets up the spread on the main form.
'
'Author : Ram Mareddy
'-------------------------------------------------------------------------------
'
Sub SetSpread ()


    '------------
    'Declarations
    '------------

    'Row heights and column widths.
    Const HEADING_ROW_HEIGHT = 20
    Const FACET_COLUMN_WIDTH = 17.9
    Const VOCABULARY_COLUMN_WIDTH = 17.9


    '-----------------------------
    'Row Heights and column widths
```

```
    '-----------------------------
    'Heading row.
    sprdSearch.RowHeight(HEADING_ROW_NUMBER) = HEADING_ROW_HEIGHT

    'Column widths
    sprdSearch.ColWidth(FACET_COLUMN_NUMBER) = FACET_COLUMN_WIDTH
    sprdSearch.ColWidth(VOCABULARY_COLUMN_NUMBER) = VOCABULARY_COLUMN_WIDTH

    '-------------
    'Column Titles
    '-------------
    sprdSearch.FontSize = gudtRwfIni.nFontSize
    sprdSearch.FontName = gudtRwfIni.sFontName
    sprdSearch.FontBold = gudtRwfIni.bFontBold

    SpreadSetText sprdSearch, FACET_COLUMN_NUMBER, HEADING_ROW_NUMBER, "Facet"
    SpreadSetText sprdSearch, VOCABULARY_COLUMN_NUMBER, HEADING_ROW_NUMBER, "Vocabulary"


    '-----------------
    'Column cell types
    '-----------------
    'frmmain.sprBoard.Col = COLUMN_RETURNS
    'frmmain.sprBoard.Row = SS_ALL_ROWS
    'frmmain.sprBoard.CellType = SS_CELL_TYPE_TIME
    'frmmain.sprBoard.TypeTime24Hour = SS_CELL_TIME_12_HOUR_CLOCK
    'frmmain.sprBoard.TypeTimeSeconds = False
    'frmmain.sprBoard.TypeSpin = True
    'frmmain.sprBoard.TypeHAlign = SS_CELL_H_ALIGN_LEFT
    'frmmain.sprBoard.TypeTimeSeparator = 58
    'frmmain.sprBoard.TypeTimeMin = "000000"
    'frmmain.sprBoard.TypeTimeMax = "235959"


    '------------------
    'Locking the Spread
    '------------------
    'To start with lock the whole spread.
    'frmmain.sprBoard.Col = SS_ALL_COLUMNS
    'frmmain.sprBoard.Row = SS_ALL_ROWS
    'frmmain.sprBoard.Lock = True


    '------------------
    'Column Appearances
    '------------------
    sprdSearch.Row = SS_ALL_ROWS
    sprdSearch.Col = VOCABULARY_COLUMN_NUMBER
    sprdSearch.TypeHAlign = SS_CELL_H_ALIGN_RIGHT


End Sub

'-------------------------------------------------------------------------------
'Name: sprdSearch_Click
'
'Type: Subroutine
'
'Parameters: Col is the column of the selected cell, Row is the row of the
'selected cell.
'
'Returns: None
'
'Description: If the cell is in the vocabulary column, then put a combo box
'with the related vocabulary as contents.
'
'Author: Ram Mareddy
```

```
'------------------------------------------------------------------------------
'
Sub sprdSearch_Click (Col As Long, Row As Long)

    '---------------------
    'Variable Declarations
    '---------------------
    'Column which contains the vocabulary terms.
    Const VOCABULARY_COLUMN_NUMBER = 1
    Const SYNONYM_COLUMN_NUMBER = 2

    '---------------------
    'Combo with vocabulary
    '---------------------
    If Col = VOCABULARY_COLUMN_NUMBER Then
        FillSearchCombo Col, Row

    ElseIf Col = SYNONYM_COLUMN_NUMBER Then
        FillSynonymCombo Col, Row

    End If


End Sub

'------------------------------------------------------------------------------
'Name: sprdSearch_LeaveCell
'
'Type: Subroutine
'
'Parameters: Col is the column of the selected cell, Row is the row of the
'selected cell, NewCol is the column of the cell to which the selection is
'moving to and NewRow is row of the same cell.
'
'Returns: None
'
'Description: If the cell is in the vocabulary column, then take out the
'combo box that is presently in that cell.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------------
'
Sub sprdSearch_LeaveCell (Col As Long, Row As Long, NewCol As Long, NewRow As Long, Cancel
As Integer)

    '---------------------
    'Variable Declarations
    '---------------------
    'Column which contains the vocabulary terms.
    Const VOCABULARY_COLUMN_NUMBER = 1
    Const SYNONYM_COLUMN_NUMBER = 2

    '----------------------
    'Make the cell type text
    '----------------------
    If (Col = VOCABULARY_COLUMN_NUMBER) Or (Col = SYNONYM_COLUMN_NUMBER) Then
        sprdSearch.CellType = SS_CELL_TYPE_STATIC_TEXT
        sprdSearch.TypeHAlign = SS_CELL_H_ALIGN_RIGHT
    End If

End Sub

Sub SynonymPreparation ()

    '-----------------------------------------------
    'Reset the spread to accomodate the synonym column
    '-----------------------------------------------
```

```
    ResetSpread


    '---------------------------------
    'Change the title of the search box
    '---------------------------------
    frmSearch.Caption = "Search With Synonyms"

End Sub

'frmThesaurus
VERSION 2.00
Begin Form frmThesaurus
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Thesaurus"
   ClientHeight    =   5175
   ClientLeft      =   360
   ClientTop       =   2715
   ClientWidth     =   8295
   Height          =   5580
   Left            =   300
   LinkTopic       =   "Form2"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   ScaleHeight     =   5175
   ScaleWidth      =   8295
   Top             =   2370
   Width           =   8415
   Begin SSPanel Panel3D1
      BevelWidth      =   3
      BorderWidth     =   1
      Height          =   5175
      Left            =   0
      TabIndex        =   8
      Top             =   0
      Width           =   8295
      Begin SSFrame Frame3D2
         Caption         =   "&Commands"
         ForeColor       =   &H00000000&
         Height          =   3735
         Left            =   6360
         ShadowStyle     =   1  'Raised
         TabIndex        =   15
         Top             =   240
         Width           =   1695
         Begin SSCommand cmd3dAdd
            AutoSize        =   2  'Adjust Button Size To Picture
            Caption         =   "&Add"
            Height          =   975
            Left            =   360
            Picture         =   RWFTHES.FRX:0000
            TabIndex        =   5
            Top             =   480
            Width           =   975
         End
         Begin SSCommand cmd3dRemove
            AutoSize        =   2  'Adjust Button Size To Picture
            Caption         =   "&Remove"
            Height          =   975
            Left            =   360
            Picture         =   RWFTHES.FRX:0302
            TabIndex        =   6
            Top             =   1440
            Width           =   975
         End
         Begin SSCommand cmd3dDone
            AutoSize        =   2  'Adjust Button Size To Picture
```

```
      Caption         =   "&Done"
      Height          =   975
      Left            =   360
      Picture         =   RWFTHES.FRX:0604
      TabIndex        =   7
      Top             =   2400
      Width           =   975
   End
End
Begin SSFrame Frame3D1
   Caption         =   "Thesaurus"
   ForeColor       =   &H00000000&
   Height          =   4695
   Left            =   240
   ShadowStyle     =   1  'Raised
   TabIndex        =   9
   Top             =   240
   Width           =   5655
   Begin SSPanel Panel3D3
      AutoSize        =   3  'AutoSize Child To Panel
      BevelOuter      =   1  'Inset
      BevelWidth      =   3
      BorderWidth     =   1
      Height          =   1380
      Left            =   720
      TabIndex        =   12
      Top             =   3000
      Width           =   4695
      Begin ListBox lstWordSynonyms
         FontBold        =   -1  'True
         FontItalic      =   0   'False
         FontName        =   "Courier New"
         FontSize        =   8.25
         FontStrikethru  =   0   'False
         FontUnderline   =   0   'False
         Height          =   1290
         Left            =   45
         MultiSelect     =   1  'Simple
         TabIndex        =   4
         Top             =   45
         Width           =   4605
      End
   End
   Begin SSOption opt3dRemoveThesaurus
      Caption         =   "&Remove from Thesaurus"
      Height          =   375
      Left            =   480
      TabIndex        =   3
      TabStop         =   0   'False
      Top             =   2520
      Width           =   2415
   End
   Begin SSPanel Panel3D2
      AutoSize        =   3  'AutoSize Child To Panel
      BevelOuter      =   1  'Inset
      BevelWidth      =   3
      BorderWidth     =   1
      Height          =   375
      Left            =   720
      TabIndex        =   11
      Top             =   960
      Width           =   2055
      Begin TextBox txtWord
         Height          =   285
         Left            =   45
         TabIndex        =   1
         Top             =   45
```

```
            Width           =   1965
         End
      End
      Begin SSOption opt3dAddThesaurus
         Caption         =   "&Add to Thesaurus"
         Height          =   255
         Left            =   480
         TabIndex        =   0
         Top             =   360
         Width           =   1935
      End
      Begin SSPanel Panel3D4
         AutoSize        =   3  'AutoSize Child To Panel
         BevelOuter      =   1  'Inset
         BevelWidth      =   3
         BorderWidth     =   1
         Height          =   1335
         Left            =   3000
         TabIndex        =   10
         Top             =   960
         Width           =   2415
         Begin TextBox txtSynonyms
            Height          =   1245
            Left            =   45
            MultiLine       =   -1  'True
            TabIndex        =   2
            Top             =   45
            Width           =   2325
         End
      End
      Begin Label Label1
         BackColor       =   &H00C0C0C0&
         Caption         =   "Word:"
         Height          =   255
         Left            =   720
         TabIndex        =   14
         Top             =   720
         Width           =   855
      End
      Begin Label Label2
         BackColor       =   &H00C0C0C0&
         Caption         =   "Synonyms:"
         Height          =   255
         Left            =   3000
         TabIndex        =   13
         Top             =   720
         Width           =   1095
      End
   End
End
Option Explicit

'--------------------------------------------------------------------------
'Name: AddThesaurus
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This routine adds a thesaurs entry to the database.
'--------------------------------------------------------------------------
'
Sub AddThesaurus ()
```

```
'--------------------
'Variable Declaration
'--------------------
'Facet information.
Dim sWordName As String
Dim sSynonyms As String

'Qurey to obtain the word ids.
Dim sqryWordId As String
Dim sWordId As String
Dim nMaxWordId As Integer
Dim nWordId As Integer

'Dynaset containing the word ids.
Dim dsWordId As Dynaset

'Message box variables.
Dim sMessage As String
Dim sTitle As String
Dim nResponse As Integer

'Thesaurus table.
Dim tblWord As table

'--------------
'Check Text Box
'--------------
If (txtWord.Text = "") Then
    sMessage = "Name box is empty. Null word can not be added."
    sTitle = "Add Word"
    nResponse = MsgBox(sMessage, MB_OK + MB_ICONEXCLAMATION, sTitle)
    Exit Sub
End If


'---------------------------------
'Word and the synonyms to be added
'---------------------------------
sWordName = txtWord.Text
sSynonyms = txtSynonyms.Text


'------------------
'Get the maximum-id
'------------------
sqryWordId = "SELECT WordId FROM THESAURUS"
Set dsWordId = gdbRwf.CreateDynaset(sqryWordId)

Do Until dsWordId.EOF
    sWordId = dsWordId("WordId")
    If nMaxWordId < CInt(sWordId) Then
        nMaxWordId = CInt(sWordId)
    End If
    dsWordId.MoveNext
Loop

nWordId = nMaxWordId + 1


'--------------
'Add New Record
'--------------
Set tblWord = gdbRwf.OpenTable("THESAURUS")

tblWord.AddNew
tblWord("Word") = sWordName
tblWord("SynonymString") = sSynonyms
```

```
    tblWord("WordId") = nWordId
    tblWord.Update
    tblWord.Close

End Sub

Sub cmd3dAdd_Click ()


    '--------------------
    'Add Word to Database
    '--------------------
    AddThesaurus

    '--------------------
    'Update all the boxes
    '--------------------
    FillBoxes

    '---------------------
    'Set focus to Name Box
    '---------------------
    txtWord.SetFocus


End Sub

Sub cmd3dDone_Click ()

    '---------------------
    'Variable Declarations
    '---------------------
    'Message box variables.
    Dim sMessage As String
    Dim sTitle As String
    Dim nResponse As Integer

    'Check the Text Box to see the user is mistakenly closing.
    If txtWord.Text <> "" Then

        sMessage = "Name box is not empty, may be you are trying to add a word."
        sMessage = sMessage + "  Do you want to go back?"
        sTitle = "Add Word Message"

        nResponse = MsgBox(sMessage, MB_YESNO, sTitle)

        'if user wants to go back, exit this subroutine.
        If nResponse = IDYES Then
            'Give focus to text box.
            EnableAddThesaurus
            txtWord.SetFocus

            Exit Sub
        End If

      End If



    Unload Me

End Sub

Sub cmd3dRemove_Click ()

    RemoveWord
```

```
End Sub

'-------------------------------------------------------------------------
'Name: EnableAddThesaurus
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function enables all the add thesaurus boxes and buttons,
'where as the EnableRemoveThesaurus does the vice versa.
'-------------------------------------------------------------------------
'
Sub EnableAddThesaurus ()

    '------
    'Enable
    '------
    txtWord.Enabled = True
    txtSynonyms.Enabled = True
    cmd3dAdd.Enabled = True

    '-------
    'Disable
    '-------
    opt3dRemoveThesaurus.Value = False
    lstWordSynonyms.Enabled = False
    cmd3dRemove.Enabled = False

End Sub

'-------------------------------------------------------------------------
'Name: EnableRemoveThesaurus
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function enables all the remove thesaurus boxes and buttons,
'where as the EnableAddThesaurus does the vice versa.
'-------------------------------------------------------------------------
'
Sub EnableRemoveThesaurus ()

    '------
    'Enable
    '------
    lstWordSynonyms.Enabled = True
    cmd3dRemove.Enabled = True

    '-------
    'Disable
    '-------
    opt3dAddThesaurus.Value = False
    txtWord.Enabled = False
    txtSynonyms.Enabled = False
    cmd3dAdd.Enabled = False

End Sub

'-------------------------------------------------------------------------
'Name: FillBoxes
```

```
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This subroutine fills all the boxes, i.e. text box, listbox
'with information. This can also be called after ADD or REMOVE is done.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
'
Sub FillBoxes ()

    Dim sqryWord As String
    Dim sWordSynonyms As String
    Dim dsWord As Dynaset


    '---------------------
    'Get Word and Synonyms
    '---------------------
    sqryWord = "SELECT Word, SynonymString FROM THESAURUS"
    Set dsWord = gdbRwf.CreateDynaset(sqryWord)

    '--------------
    'Empty Name Box
    '--------------
    txtWord.Text = ""
    txtSynonyms.Text = ""


    '--------------------------
    'Fill Word+Synonyms List Box
    '--------------------------
    'First clear the list box.
    lstWordSynonyms.Clear

    Do Until dsWord.EOF
        sWordSynonyms = "(" + dsWord("Word") + ")"
        sWordSynonyms = sWordSynonyms + Space$(20 - Len(sWordSynonyms)) + " : "
        sWordSynonyms = sWordSynonyms + "(" + dsWord("SynonymString") + ")"
        lstWordSynonyms.AddItem sWordSynonyms
        dsWord.MoveNext
    Loop

End Sub

Sub Form_Load ()

    '---------------
    'Center the form
    '---------------
    CenterForm Me

    '---------------------------------
    'Fill boxes on the form with details
    '---------------------------------
    FillBoxes


End Sub

Sub opt3dAddThesaurus_Click (Value As Integer)

    EnableAddThesaurus
```

```
    '--------------------
    'Set focus to Name Box
    '--------------------
    txtWord.SetFocus


End Sub

Sub opt3dRemoveThesaurus_Click (Value As Integer)

    EnableRemoveThesaurus


End Sub

'------------------------------------------------------------------------
'Name: RemoveWord
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function removes one or more words selected in the
'remove list box.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub RemoveWord ()

    '--------------------
    'Variable Declarations
    '--------------------
    'Loop counter.
    Dim i As Integer

    'Full string in the remove box.
    Dim sWordSynonyms As String

    'Name of the word.
    Dim sWord As String

    'Start and end for string parsing.
    Dim nStart As Integer
    Dim nLength As Integer

    'Query to remove the word.
    Dim sqryRemoveWord As String


    '---------------------------------------
    'All the items selected should be removed
    '---------------------------------------
    For i = 0 To lstWordSynonyms.ListCount - 1
        If lstWordSynonyms.Selected(i) = True Then
            sWordSynonyms = lstWordSynonyms.List(i)

            'Parse the sWordSynonyms into word and synonyms

            'Format of name is (word)   : (synonyms), length of name is position 2 to
            'the character before ).
            nStart = 2
            nLength = InStr(1, sWordSynonyms, ")") - nStart
            sWord = Mid$(sWordSynonyms, nStart, nLength)
```

```
            sqryRemoveWord = "DELETE FROM THESAURUS WHERE Word = '" + sWord + "'"
            gdbRwf.Execute (sqryRemoveWord)
        End If
    Next


    '------------------
    'Refresh boxes again
    '------------------
    FillBoxes

End Sub

'frmArtifactusage
VERSION 2.00
Begin Form frmArtifactusage
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Usage Report By Artifact"
   ClientHeight    =   4095
   ClientLeft      =   1470
   ClientTop       =   2475
   ClientWidth     =   7335
   Height          =   4500
   Left            =   1410
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   ScaleHeight     =   4095
   ScaleWidth      =   7335
   Top             =   2130
   Width           =   7455
   Begin SSPanel Panel3D1
      BevelWidth      =   2
      Height          =   4095
      Left            =   0
      TabIndex        =   0
      Top             =   0
      Width           =   7335
      Begin SSFrame Frame3D3
         Caption         =   "&Commands"
         ForeColor       =   &H00000000&
         Height          =   1815
         Left            =   5520
         ShadowStyle     =   1  'Raised
         TabIndex        =   3
         Top             =   240
         Width           =   1575
         Begin SSCommand cmd3dDone
            AutoSize        =   2  'Adjust Button Size To Picture
            Caption         =   "&Done"
            Height          =   975
            Left            =   360
            Picture         =   RWFUSAGE.FRX:0000
            TabIndex        =   4
            Top             =   480
            Width           =   975
         End
      End
      Begin SSFrame Frame3D1
         Caption         =   "Usage Report By Artifact"
         ForeColor       =   &H00000000&
         Height          =   3615
         Left            =   240
         ShadowStyle     =   1  'Raised
         TabIndex        =   1
         Top             =   240
```

```
         Width           =   5055
         Begin SpreadSheet sprdUsage
            AllowResize     =   -1  'True
            FontBold        =   -1  'True
            FontItalic      =   0   'False
            FontName        =   "MS Sans Serif"
            FontSize        =   8.25
            FontStrikethru  =   0   'False
            FontUnderline   =   0   'False
            Height          =   3015
            InterfaceDesigner=   RWFUSAGE.FRX:0302
            Left            =   240
            MaxCols         =   1
            MaxRows         =   20
            TabIndex        =   2
            Top             =   360
            Width           =   4575
         End
      End
   End
End
Option Explicit
'Row and column numbers.
Const HEADING_ROW_NUMBER = 0
Const NAME_COLUMN_NUMBER = 0
Const USAGE_COLUMN_NUMBER = 1

Sub cmd3dDone_Click ()

    Unload Me

End Sub

'--------------------------------------------------------------------------------
'Name: FillSpread
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: Spreadsheet in the current form is filled with the requested
'information.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------------
'
Sub FillSpread ()

    '---------------------
    'Variable Decalrations
    '---------------------
    'Queries.
    Dim sqryArtifact As String
    Dim sqryUsage As String

    'Dynasets.
    Dim dsArtifact As Dynaset
    Dim dsUsage As Dynaset

    'Usage and row
    Dim nSpreadRow As Integer
    Dim nNumRecords As Integer
    Dim nCurrentArtifactId As Integer
```

```
    '------------------------------
    'Get the name and id of the Artifact
    '------------------------------
    sqryArtifact = "SELECT ArtifactId, ArtifactFileName FROM Artifact"
    Set dsArtifact = gdbRwf.CreateDynaset(sqryArtifact)

    'Get the record count.
    Do Until dsArtifact.EOF
        nNumRecords = nNumRecords + 1
        dsArtifact.MoveNext
    Loop

    sprdUsage.MaxRows = nNumRecords

    nSpreadRow = 1
    dsArtifact.MoveFirst
    Do Until dsArtifact.EOF
        'Put name on the spread.
        sprdUsage.Row = nSpreadRow
        sprdUsage.Col = NAME_COLUMN_NUMBER
        sprdUsage.Text = dsArtifact("ArtifactFileName")

        'Get the number of events.
        nCurrentArtifactId = dsArtifact("ArtifactId")
        sqryUsage = "SELECT ArtifactId FROM EVENT WHERE ArtifactId = " +
CStr(nCurrentArtifactId)
        Set dsUsage = gdbRwf.CreateDynaset(sqryUsage)

        nNumRecords = 0
        Do Until dsUsage.EOF
            nNumRecords = nNumRecords + 1
            dsUsage.MoveNext
        Loop

        'Put the usage.
        sprdUsage.Row = nSpreadRow
        sprdUsage.Col = USAGE_COLUMN_NUMBER
        sprdUsage.Text = nNumRecords

        dsArtifact.MoveNext
        nSpreadRow = nSpreadRow + 1

    Loop


End Sub

Sub Form_Load ()

    '---------------
    'Center the form
    '---------------
    CenterForm Me


    '----------
    'Set spread
    '----------
    SetSpread


    '-----------
    'Fill Spread
    '-----------
    FillSpread
```

```
End Sub

'--------------------------------------------------------------------------------
'Name: SetSpread
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function sets up the spread on the usage report form.
'
'Author : Ram Mareddy
'--------------------------------------------------------------------------------
'
Sub SetSpread ()

    '------------
    'Declarations
    '------------

    'Row heights and column widths.
    Const HEADING_ROW_HEIGHT = 20
    Const NAME_COLUMN_WIDTH = 17.9
    Const USAGE_COLUMN_WIDTH = 17.9


    '-----------------------------
    'Row Heights and column widths
    '-----------------------------
    'Heading row.
    sprdUsage.RowHeight(HEADING_ROW_NUMBER) = HEADING_ROW_HEIGHT

    'Column widths
    sprdUsage.ColWidth(NAME_COLUMN_NUMBER) = NAME_COLUMN_WIDTH
    sprdUsage.ColWidth(USAGE_COLUMN_NUMBER) = USAGE_COLUMN_WIDTH

    '-------------
    'Column Titles
    '-------------
    sprdUsage.FontSize = gudtRwfIni.nFontSize
    sprdUsage.FontName = gudtRwfIni.sFontName
    sprdUsage.FontBold = gudtRwfIni.bFontBold

    SpreadSetText sprdUsage, NAME_COLUMN_NUMBER, HEADING_ROW_NUMBER, "Name"
    SpreadSetText sprdUsage, USAGE_COLUMN_NUMBER, HEADING_ROW_NUMBER, "Usage"


    '------------------
    'Column Appearances
    '------------------
    sprdUsage.Row = SS_ALL_ROWS
    sprdUsage.Col = USAGE_COLUMN_NUMBER
    sprdUsage.TypeHAlign = SS_CELL_H_ALIGN_RIGHT


End Sub

'frmUser
VERSION 2.00
Begin Form frmUser
   BorderStyle     =   1  'Fixed Single
   Caption         =   "User"
   ClientHeight    =   5520
   ClientLeft      =   165
```

```
ClientTop       =   2145
ClientWidth     =   8280
Height          =   5925
Left            =   105
LinkTopic       =   "Form1"
MaxButton       =   0   'False
MinButton       =   0   'False
ScaleHeight     =   5520
ScaleWidth      =   8280
Top             =   1800
Width           =   8400
Begin SSPanel Panel3D1
   BevelWidth       =   3
   BorderWidth      =   1
   Height           =   5535
   Left             =   0
   TabIndex         =   11
   Top              =   0
   Width            =   8295
   Begin SSFrame Frame3D1
      Caption          =   "User"
      ForeColor        =   &H00000000&
      Height           =   5055
      Left             =   240
      ShadowStyle      =   1  'Raised
      TabIndex         =   14
      Top              =   240
      Width            =   5655
      Begin SSPanel Panel3D8
         AutoSize         =   3  'AutoSize Child To Panel
         BevelOuter       =   1  'Inset
         BevelWidth       =   3
         BorderWidth      =   1
         Height           =   1815
         Left             =   3000
         TabIndex         =   27
         Top              =   1680
         Width            =   2415
         Begin TextBox txtUserComments
            Height           =   1725
            Left             =   45
            MultiLine        =   -1  'True
            TabIndex         =   6
            Top              =   45
            Width            =   2325
         End
      End
      Begin SSPanel Panel3D7
         AutoSize         =   3  'AutoSize Child To Panel
         BevelOuter       =   1  'Inset
         BevelWidth       =   3
         BorderWidth      =   1
         Height           =   390
         Left             =   720
         TabIndex         =   25
         Top              =   3120
         Width            =   2055
         Begin ComboBox cboRole
            Height           =   300
            Left             =   45
            Style            =   2  'Dropdown List
            TabIndex         =   5
            Top              =   45
            Width            =   1965
         End
      End
      Begin SSPanel Panel3D6
```

```
   AutoSize          =   3  'AutoSize Child To Panel
   BevelOuter        =   1  'Inset
   BevelWidth        =   3
   BorderWidth       =   1
   Height            =   375
   Left              =   720
   TabIndex          =   23
   Top               =   2400
   Width             =   2055
   Begin TextBox txtUserTelephone
      Height            =   285
      Left              =   45
      TabIndex          =   4
      Top               =   45
      Width             =   1965
   End
End
Begin SSPanel Panel3D5
   AutoSize          =   3  'AutoSize Child To Panel
   BevelOuter        =   1  'Inset
   BevelWidth        =   3
   BorderWidth       =   1
   Height            =   375
   Left              =   720
   TabIndex          =   22
   Top               =   1680
   Width             =   2055
   Begin TextBox txtUserAddress
      Height            =   285
      Left              =   45
      TabIndex          =   3
      Top               =   45
      Width             =   1965
   End
End
Begin SSPanel Panel3D4
   AutoSize          =   3  'AutoSize Child To Panel
   BevelOuter        =   1  'Inset
   BevelWidth        =   3
   BorderWidth       =   1
   Height            =   375
   Left              =   3000
   TabIndex          =   17
   Top               =   960
   Width             =   2415
   Begin TextBox txtUserLastName
      Height            =   285
      Left              =   45
      TabIndex          =   2
      Top               =   45
      Width             =   2325
   End
End
Begin SSOption opt3dAddUser
   Caption           =   "&Add User"
   Height            =   255
   Left              =   480
   TabIndex          =   0
   Top               =   360
   Width             =   1935
End
Begin SSPanel Panel3D2
   AutoSize          =   3  'AutoSize Child To Panel
   BevelOuter        =   1  'Inset
   BevelWidth        =   3
   BorderWidth       =   1
   Height            =   375
```

```
   Left            =   720
   TabIndex        =   16
   Top             =   960
   Width           =   2055
   Begin TextBox txtUserFirstName
      Height          =   285
      Left            =   45
      TabIndex        =   1
      Top             =   45
      Width           =   1965
   End
End
Begin SSOption opt3dRemoveUser
   Caption         =   "&Remove User"
   Height          =   375
   Left            =   480
   TabIndex        =   7
   TabStop         =   0   'False
   Top             =   3600
   Width           =   2415
End
Begin SSPanel Panel3D3
   AutoSize        =   3   'AutoSize Child To Panel
   BevelOuter      =   1   'Inset
   BevelWidth      =   3
   BorderWidth     =   1
   Height          =   960
   Left            =   720
   TabIndex        =   15
   Top             =   3960
   Width           =   4695
   Begin ListBox lstName
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "Courier New"
      FontSize        =   8.25
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      Height          =   870
      Left            =   45
      MultiSelect     =   1   'Simple
      TabIndex        =   8
      Top             =   45
      Width           =   4605
   End
End
Begin Label Label6
   BackColor       =   &H00C0C0C0&
   Caption         =   "Comments:"
   Height          =   255
   Left            =   3000
   TabIndex        =   26
   Top             =   1440
   Width           =   1095
End
Begin Label Label5
   BackColor       =   &H00C0C0C0&
   Caption         =   "Role:"
   Height          =   255
   Left            =   720
   TabIndex        =   24
   Top             =   2880
   Width           =   1095
End
Begin Label Label4
   BackColor       =   &H00C0C0C0&
   Caption         =   "Telephone:"
```

```
      Height          =   255
      Left            =   720
      TabIndex        =   21
      Top             =   2160
      Width           =   1095
   End
   Begin Label Label3
      BackColor       =   &H00C0C0C0&
      Caption         =   "Address:"
      Height          =   255
      Left            =   720
      TabIndex        =   20
      Top             =   1440
      Width           =   1095
   End
   Begin Label Label2
      BackColor       =   &H00C0C0C0&
      Caption         =   "Last Name:"
      Height          =   255
      Left            =   3000
      TabIndex        =   19
      Top             =   720
      Width           =   1095
   End
   Begin Label Label1
      BackColor       =   &H00C0C0C0&
      Caption         =   "First Name:"
      Height          =   255
      Left            =   720
      TabIndex        =   18
      Top             =   720
      Width           =   1095
   End
End
Begin SSFrame Frame3D2
   Caption         =   "&Commands"
   ForeColor       =   &H00000000&
   Height          =   3735
   Left            =   6360
   ShadowStyle     =   1  'Raised
   TabIndex        =   12
   Top             =   240
   Width           =   1695
   Begin SSCommand cmd3dDone
      AutoSize        =   2  'Adjust Button Size To Picture
      Caption         =   "&Done"
      Height          =   975
      Left            =   360
      Picture         =   RWFUSER.FRX:0000
      TabIndex        =   13
      Top             =   2400
      Width           =   975
   End
   Begin SSCommand cmd3dRemove
      AutoSize        =   2  'Adjust Button Size To Picture
      Caption         =   "&Remove"
      Height          =   975
      Left            =   360
      Picture         =   RWFUSER.FRX:0302
      TabIndex        =   10
      Top             =   1440
      Width           =   975
   End
   Begin SSCommand cmd3dAdd
      AutoSize        =   2  'Adjust Button Size To Picture
      Caption         =   "&Add"
      Height          =   975
```

```
            Left            =   360
            Picture         =   RWFUSER.FRX:0604
            TabIndex        =   9
            Top             =   480
            Width           =   975
         End
      End
   End
End
Option Explicit

'--------------------------------------------------------------------------------
'Name: AddUser
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This routine adds a user to the database.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------------
'
Sub AddUser ()


    '--------------------
    'Variable Declaration
    '--------------------
    'User information.
    Dim sRoleName As String
    Dim sUserFirstName As String
    Dim sUserLastName As String
    Dim sUserAddress As String
    Dim sUserTelephone As String

    'Query and dynaset.
    Dim sqryUserId As String
    Dim dsUserId As Dynaset
    Dim tblUser As Table

    'Ids.
    Dim nRoleId As Integer
    Dim sUserId As String
    Dim nUserId As Integer
    Dim nMaxUserId As Integer

    'Message box variables.
    Dim sMessage As String
    Dim sTitle As String
    Dim nResponse As Integer


    '--------------
    'Check Text Box
    '--------------
    If (txtUserFirstName.Text = "") Then
        sMessage = "Name box is empty. Null user can not be added."
        sTitle = "Add User"
        nResponse = MsgBox(sMessage, MB_OK + MB_ICONEXCLAMATION, sTitle)
        Exit Sub
    End If

    '----------
    'Get RoleId
```

```
    '----------
    sRoleName = cboRole.Text
    nRoleId = GetRoleId(sRoleName)
    sUserFirstName = txtUserFirstName.Text
    sUserLastName = txtUserLastName.Text
    sUserAddress = txtUserAddress.Text
    sUserTelephone = txtUserTelephone.Text

    '----------
    'Get UserId
    '----------
    sqryUserId = "SELECT UserId FROM USERS"
    Set dsUserId = gdbRwf.CreateDynaset(sqryUserId)

    Do Until dsUserId.EOF
        sUserId = dsUserId("UserId")
        If nMaxUserId < CInt(sUserId) Then
            nMaxUserId = CInt(sUserId)
        End If
        dsUserId.MoveNext
    Loop

    nUserId = nMaxUserId + 1


    '--------------
    'Add New Record
    '--------------
    Set tblUser = gdbRwf.OpenTable("USERS")

    tblUser.AddNew
    tblUser("UserId") = nUserId
    tblUser("RoleId") = nRoleId
    tblUser("UserFirstName") = sUserFirstName
    tblUser("UserLastName") = sUserLastName
    tblUser("UserAddress") = sUserAddress
    tblUser("UserTelephone") = sUserTelephone
    tblUser.Update
    tblUser.Close

End Sub

Sub cmd3dAdd_Click ()

    '--------------------
    'Add User to Database
    '--------------------
    AddUser

    '--------------------
    'Update all the boxes
    '--------------------
    FillBoxes

    '---------------------
    'Set focus to Name Box
    '---------------------
    txtUserFirstName.SetFocus


End Sub

Sub cmd3dDone_Click ()

    '---------------------
    'Variable Declarations
    '---------------------
```

```
    'Message box.
    Dim sMessage As String
    Dim sTitle As String
    Dim nResponse As Integer

    'Check the Text Box to see the user is mistakenly closing.
    If txtUserFirstName.Text <> "" Then

        sMessage = "Name box is not empty, may be you are trying to add a user."
        sMessage = sMessage + "  Do you want to go back?"
        sTitle = "Add User Message"

        nResponse = MsgBox(sMessage, MB_YESNO, sTitle)

        'if user wants to go back, exit this subroutine.
        If nResponse = IDYES Then
            'Give focus to text box.
            EnableAddUser
            txtUserFirstName.SetFocus

            Exit Sub
        End If

      End If

    Unload Me

End Sub

Sub cmd3dRemove_Click ()

    RemoveUser


End Sub

'------------------------------------------------------------------------
'Name: EnableAddUser
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function enables all the add user boxes and buttons,
'where as the EnableRemoveUser does the vice versa.
'------------------------------------------------------------------------
'
Sub EnableAddUser ()

    '------
    'Enable
    '------
    txtUserFirstName.Enabled = True
    txtUserLastName.Enabled = True
    txtUserAddress.Enabled = True
    txtUserTelephone.Enabled = True
    cboRole.Enabled = True
    cmd3dAdd.Enabled = True

    '-------
    'Disable
    '-------
    opt3dRemoveUser.Value = False
    lstName.Enabled = False
    cmd3dRemove.Enabled = False
```

```
End Sub

'------------------------------------------------------------------------
'Name: EnableRemoveUser
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function enables all the remove user boxes and buttons,
'where as the EnableAddUser does the vice versa.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub EnableRemoveUser ()

    '------
    'Enable
    '------
    lstName.Enabled = True
    cmd3dRemove.Enabled = True

    '-------
    'Disable
    '-------
    opt3dAddUser.Value = False
    txtUserFirstName.Enabled = False
    txtUserLastName.Enabled = False
    txtUserAddress.Enabled = False
    txtUserTelephone.Enabled = False
    cboRole.Enabled = False
    cmd3dAdd.Enabled = False

End Sub

'------------------------------------------------------------------------
'Name: FillBoxes
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This subroutine fills all the boxes, i.e. text box, listbox
'with information. This can also be called after ADD or REMOVE is done.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub FillBoxes ()

    '---------------------
    'Variable Declarations
    '---------------------
    'Query strings
    Dim sqryName As String
    Dim sqryRole As String
    Dim sName As String

    'Dynasets.
    Dim dsName As Dynaset
    Dim dsRole As Dynaset
```

```
    'Combo box index.
    Dim nListIndex As Integer


    '--------------------------
    'Get FirstName and LastName
    '--------------------------
    sqryName = "SELECT UserFirstName, UserLastName FROM USERS"
    Set dsName = gdbRwf.CreateDynaset(sqryName)

    '---------
    'Get Roles
    '---------
    sqryRole = "SELECT RoleName FROM ROLE"
    Set dsRole = gdbRwf.CreateDynaset(sqryRole)

    '--------------
    'Empty Name Box
    '--------------
    txtUserFirstName.Text = ""
    txtUserLastName.Text = ""
    txtUserAddress.Text = ""
    txtUserTelephone.Text = ""
    txtUserComments.Text = ""

    '-------------------
    'Fill Role Combo Box
    '-------------------
    'If this is not the first time, get the list index.
    nListIndex = cboRole.ListIndex

    'Clear the combo box.
    cboRole.Clear

    Do Until dsRole.EOF
        cboRole.AddItem dsRole("RoleName")
        dsRole.MoveNext
    Loop

    'First time list index will be -1, then change it to 0.
    If nListIndex = -1 Then nListIndex = 0
    cboRole.ListIndex = nListIndex

    '--------------------------------
    'Fill LastName+FirstName List Box
    '--------------------------------
    'First clear the list box.
    lstName.Clear

    Do Until dsName.EOF
        sName = "(" + dsName("UserLastName") + ")"
        sName = sName + Space$(20 - Len(sName)) + " , "
        sName = sName + "(" + dsName("UserFirstName") + ")"
        lstName.AddItem sName
        dsName.MoveNext
    Loop

End Sub

Sub Form_Load ()

    '---------------
    'Center the form
    '---------------
    CenterForm Me
```

```
    '---------------------------------
    'Fill boxes on the form with details
    '---------------------------------
    FillBoxes


End Sub

Sub opt3dAddUser_Click (Value As Integer)

    EnableAddUser

    '--------------------
    'Set focus to Name Box
    '--------------------
    txtUserFirstName.SetFocus


End Sub

Sub opt3dRemoveUser_Click (Value As Integer)

    EnableRemoveUser

End Sub

'------------------------------------------------------------------------
'Name: RemoveUser
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function removes one or more users selected in the
'remove list box.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub RemoveUser ()

    '--------------------
    'Variable Declarations
    '--------------------
    'Name of the user.
    Dim sName As String
    Dim sFirstName As String
    Dim sLastName As String
    Dim sqryRemoveUser As String

    'Positions.
    Dim nStart As Integer
    Dim nLength As Integer
    Dim i As Integer

    '---------------------------------------
    'All the selected users should be deleted
    '---------------------------------------
    For i = 0 To lstName.ListCount - 1
        If lstName.Selected(i) = True Then
            sName = lstName.List(i)

            'Parse the sName into First Name and Last Name.

            'Format of name is (lastname)  , (fistname), length of name is position 2 to
```

```
            'the character before ).
            nStart = 2
            nLength = InStr(1, sName, ")") - nStart
            sLastName = Mid$(sName, nStart, nLength)

            'From the second ( to just before the last character.
            'Names are unique by the group.
            nStart = InStr(2, sName, "(") + 1
            nLength = Len(sName) - nStart
            sFirstName = Mid$(sName, nStart, nLength)

            sqryRemoveUser = "DELETE FROM USERS WHERE UserFirstName = '" + sFirstName + "'
AND  UserLastName = '" + sLastName + "'"
            gdbRwf.Execute (sqryRemoveUser)
        End If
    Next

    'Refresh boxes again.
    FillBoxes
End Sub

'frmVocabulary
VERSION 2.00
Begin Form frmVocabulary
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Vocabulary"
   ClientHeight    =   5175
   ClientLeft      =   1050
   ClientTop       =   3930
   ClientWidth     =   8295
   Height          =   5580
   Left            =   990
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   ScaleHeight     =   5175
   ScaleWidth      =   8295
   Top             =   3585
   Width           =   8415
   Begin SSPanel Panel3D1
      BevelWidth      =   3
      BorderWidth     =   1
      Height          =   5175
      Left            =   0
      TabIndex        =   9
      Top             =   0
      Width           =   8295
      Begin SSFrame Frame3D2
         Caption         =   "&Commands"
         ForeColor       =   &H00000000&
         Height          =   3735
         Left            =   6360
         ShadowStyle     =   1  'Raised
         TabIndex        =   16
         Top             =   240
         Width           =   1695
         Begin SSCommand cmd3dAdd
            AutoSize        =   2  'Adjust Button Size To Picture
            Caption         =   "&Add"
            Height          =   975
            Left            =   360
            Picture         =   RWFVOCAB.FRX:0000
            TabIndex        =   6
            Top             =   480
            Width           =   975
         End
         Begin SSCommand cmd3dRemove
```

```
      AutoSize        =   2  'Adjust Button Size To Picture
      Caption         =   "&Remove"
      Height          =   975
      Left            =   360
      Picture         =   RWFVOCAB.FRX:0302
      TabIndex        =   7
      Top             =   1440
      Width           =   975
   End
   Begin SSCommand cmd3dDone
      AutoSize        =   2  'Adjust Button Size To Picture
      Caption         =   "&Done"
      Height          =   975
      Left            =   360
      Picture         =   RWFVOCAB.FRX:0604
      TabIndex        =   8
      Top             =   2400
      Width           =   975
   End
End
Begin SSFrame Frame3D1
   Caption         =   "Vocabulary"
   ForeColor       =   &H00000000&
   Height          =   4695
   Left            =   240
   ShadowStyle     =   1  'Raised
   TabIndex        =   10
   Top             =   240
   Width           =   5655
   Begin SSPanel Panel3D5
      AutoSize        =   3  'AutoSize Child To Panel
      BevelOuter      =   1  'Inset
      BevelWidth      =   3
      BorderWidth     =   1
      Height          =   390
      Left            =   720
      TabIndex        =   17
      Top             =   1920
      Width           =   2055
      Begin ComboBox cboFacet
         Height          =   300
         Left            =   45
         Style           =   2  'Dropdown List
         TabIndex        =   3
         Top             =   45
         Width           =   1965
      End
   End
   Begin SSPanel Panel3D3
      AutoSize        =   3  'AutoSize Child To Panel
      BevelOuter      =   1  'Inset
      BevelWidth      =   3
      BorderWidth     =   1
      Height          =   1380
      Left            =   720
      TabIndex        =   13
      Top             =   3000
      Width           =   4695
      Begin ListBox lstNameFacet
         FontBold        =   -1  'True
         FontItalic      =   0   'False
         FontName        =   "Courier New"
         FontSize        =   8.25
         FontStrikethru  =   0   'False
         FontUnderline   =   0   'False
         Height          =   1290
         Left            =   45
```

```
      MultiSelect      =   1  'Simple
      TabIndex         =   5
      Top              =   45
      Width            =   4605
   End
End
Begin SSOption opt3dRemoveVocabulary
   Caption          =   "&Remove Vocabulary"
   Height           =   375
   Left             =   480
   TabIndex         =   4
   TabStop          =   0   'False
   Top              =   2520
   Width            =   2055
End
Begin SSPanel Panel3D2
   AutoSize         =   3  'AutoSize Child To Panel
   BevelOuter       =   1  'Inset
   BevelWidth       =   3
   BorderWidth      =   1
   Height           =   375
   Left             =   720
   TabIndex         =   12
   Top              =   960
   Width            =   2055
   Begin TextBox txtName
      Height           =   285
      Left             =   45
      TabIndex         =   1
      Top              =   45
      Width            =   1965
   End
End
Begin SSOption opt3dAddVocabulary
   Caption          =   "&Add Vocabulary"
   Height           =   255
   Left             =   480
   TabIndex         =   0
   Top              =   360
   Width            =   1815
End
Begin SSPanel Panel3D4
   AutoSize         =   3  'AutoSize Child To Panel
   BevelOuter       =   1  'Inset
   BevelWidth       =   3
   BorderWidth      =   1
   Height           =   1335
   Left             =   3000
   TabIndex         =   11
   Top              =   960
   Width            =   2415
   Begin TextBox txtComments
      Height           =   1245
      Left             =   45
      MultiLine        =   -1  'True
      TabIndex         =   2
      Top              =   45
      Width            =   2325
   End
End
Begin Label Label3
   BackColor        =   &H00C0C0C0&
   Caption          =   "Facet:"
   Height           =   255
   Left             =   720
   TabIndex         =   18
   Top              =   1680
```

```
            Width           =   1095
         End
         Begin Label Label1
            BackColor       =   &H00C0C0C0&
            Caption         =   "Name:"
            Height          =   255
            Left            =   720
            TabIndex        =   15
            Top             =   720
            Width           =   855
         End
         Begin Label Label2
            BackColor       =   &H00C0C0C0&
            Caption         =   "Comments:"
            Height          =   255
            Left            =   3000
            TabIndex        =   14
            Top             =   720
            Width           =   1095
         End
      End
   End
End
Option Explicit

'--------------------------------------------------------------------------------
'Name: AddVocabulary
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This routine adds a vocabulary term to the database.
'--------------------------------------------------------------------------------
'
Sub AddVocabulary ()

    '--------------------
    'Variable Declaration
    '--------------------
    'Vocabulary information.
    Dim sVocabularyName As String
    Dim sVocabularyComments As String
    Dim sFacetName As String

    'Qurey to obtain the vocabulary ids.
    Dim sqryVocabularyId As String
    Dim sVocabularyId As String
    Dim nMaxVocabularyId As Integer
    Dim nVocabularyId As Integer
    Dim nFacetId As Integer

    'Dynaset containing the vocabulary ids.
    Dim dsVocabularyId As dynaset

    'Message box variables.
    Dim sMessage As String
    Dim sTitle As String
    Dim nResponse As Integer

    'Table of the vocabulary.
    Dim tblVocabulary As table

    '--------------
    'Check Text Box
```

```
    '--------------
    If (txtName.Text = "") Then
        sMessage = "Name box is empty. Null vocabulary can not be added."
        sTitle = "Add Vocabulary"
        nResponse = MsgBox(sMessage, MB_OK + MB_ICONEXCLAMATION, sTitle)
        Exit Sub
    End If


    '----------------------------------------
    'Vocabulary and the description to be added
    '----------------------------------------
    sVocabularyName = txtName.Text
    sVocabularyComments = txtComments.Text


    '------------------
    'Get the maximum-id
    '------------------
    sqryVocabularyId = "SELECT VocabularyId FROM VOCABULARY"
    Set dsVocabularyId = gdbRwf.CreateDynaset(sqryVocabularyId)

    Do Until dsVocabularyId.EOF
        sVocabularyId = dsVocabularyId("VocabularyId")
        If nMaxVocabularyId < CInt(sVocabularyId) Then
            nMaxVocabularyId = CInt(sVocabularyId)
        End If
        dsVocabularyId.MoveNext
    Loop

    nVocabularyId = nMaxVocabularyId + 1


    '----------------
    'Get the Facet Id
    '----------------
    sFacetName = cboFacet.Text
    nFacetId = GetFacetId(sFacetName)

    '--------------
    'Add New Record
    '--------------
    Set tblVocabulary = gdbRwf.OpenTable("VOCABULARY")

    tblVocabulary.AddNew
    tblVocabulary("VocabularyName") = sVocabularyName
    tblVocabulary("VocabularyComments") = sVocabularyComments
    tblVocabulary("VocabularyId") = nVocabularyId
    tblVocabulary("FacetId") = nFacetId
    tblVocabulary.Update
    tblVocabulary.Close

End Sub

Sub AdjustClassification ()

End Sub

Sub cmd3dAdd_Click ()


    '--------------------------
    'Add Vocabulary to Database
    '--------------------------
    AddVocabulary

    '--------------------
```

```
    'Update all the boxes
    '--------------------
    FillBoxes

    '---------------------
    'Set focus to Name Box
    '---------------------
    txtName.SetFocus


End Sub

Sub cmd3dDone_Click ()

    '---------------------
    'Variable Declarations
    '---------------------
    'Message box variables.
    Dim sMessage As String
    Dim sTitle As String
    Dim nResponse As Integer

    'Check the Text Box to see the user is mistakenly closing.
    If txtName.Text <> "" Then

        sMessage = "Name box is not empty, may be you are trying to add a vocabulary
term."
        sMessage = sMessage + "  Do you want to go back?"
        sTitle = "Add Vocabulary Message"

        nResponse = MsgBox(sMessage, MB_YESNO, sTitle)

        'if user wants to go back, exit this subroutine.
        If nResponse = IDYES Then
            'Give focus to text box.
            EnableAddVocabulary
            txtName.SetFocus

            Exit Sub
        End If

      End If


    Unload Me

End Sub

Sub cmd3dRemove_Click ()

    RemoveVocabulary

End Sub

'-----------------------------------------------------------------------------
'Name: EnableAddVocabulary
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function enables all the add vocabulary boxes and buttons,
'where as the EnableRemoveVocabulary does the vice versa.
'-----------------------------------------------------------------------------
```

```
'
Sub EnableAddVocabulary ()

    '------
    'Enable
    '------
    txtName.Enabled = True
    txtComments.Enabled = True
    cboFacet.Enabled = True
    cmd3dAdd.Enabled = True

    '-------
    'Disable
    '-------
    opt3dRemoveVocabulary.Value = False
    lstNameFacet.Enabled = False
    cmd3dRemove.Enabled = False

End Sub

'--------------------------------------------------------------------------
'Name: EnableRemoveVocabulary
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function enables all the remove vocabulary boxes and buttons,
'where as the EnableAddVocabulary does the vice versa.
'--------------------------------------------------------------------------
'
Sub EnableRemoveVocabulary ()

    '------
    'Enable
    '------
    lstNameFacet.Enabled = True
    cmd3dRemove.Enabled = True

    '-------
    'Disable
    '-------
    opt3dAddVocabulary.Value = False
    txtName.Enabled = False
    txtComments.Enabled = False
    cboFacet.Enabled = False
    cmd3dAdd.Enabled = False

End Sub

'--------------------------------------------------------------------------
'Name: FillBoxes
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This subroutine fills all the boxes, i.e. text boxes, combobox,
'listbox, with information. This can also be called after ADD or REMOVE is
'done.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
```

```
'
Sub FillBoxes ()

    '--------------------
    'Variable Declarations
    '--------------------
    Dim sqryNameFacet As String
    Dim sqryFacet As String
    Dim sNameFacet As String

    Dim dsNameFacet As dynaset
    Dim dsFacet As dynaset
    Dim nListIndex As Integer

    '------------------
    'Get Name and Facet
    '------------------
    sqryNameFacet = "SELECT VocabularyName, FacetName FROM VOCABULARY, FACET WHERE
VOCABULARY.FacetId = FACET.FacetId"
    Set dsNameFacet = gdbRwf.CreateDynaset(sqryNameFacet)

    '--------------------
    'Get Just Facet Names
    '--------------------
    sqryFacet = "SELECT FacetName FROM FACET"
    Set dsFacet = gdbRwf.CreateDynaset(sqryFacet)

    '--------------
    'Empty Name Box
    '--------------
    txtName.Text = ""
    txtComments.Text = ""

    '--------------------
    'Fill Facet Combo Box
    '--------------------

    'If this is not the first time, get the list index.
    nListIndex = cboFacet.ListIndex

    'Clear the combo box.
    cboFacet.Clear

    Do Until dsFacet.EOF
        cboFacet.AddItem dsFacet("FacetName")
        dsFacet.MoveNext
    Loop

    'First time list index will be -1, then change it to 0.
    If nListIndex = -1 Then nListIndex = 0
    cboFacet.ListIndex = nListIndex

    '------------------------
    'Fill Name+Facet List Box
    '------------------------
    'First clear the list box.
    lstNameFacet.Clear

    Do Until dsNameFacet.EOF
        sNameFacet = "(" + dsNameFacet("VocabularyName") + ")"
        sNameFacet = sNameFacet + Space$(20 - Len(sNameFacet)) + " : "
        sNameFacet = sNameFacet + "(" + dsNameFacet("FacetName") + ")"
        lstNameFacet.AddItem sNameFacet
        dsNameFacet.MoveNext
    Loop

End Sub
```

```
Sub Form_Load ()

    '---------------
    'Center the form
    '---------------
    CenterForm Me


    '-----------------------------------
    'Fill boxes on the form with details
    '-----------------------------------
    FillBoxes

End Sub

Sub opt3dAddVocabulary_Click (Value As Integer)

    EnableAddVocabulary

    '---------------------
    'Set focus to Name Box
    '---------------------
    txtName.SetFocus


End Sub

Sub opt3dRemoveVocabulary_Click (Value As Integer)

    EnableRemoveVocabulary

End Sub

'------------------------------------------------------------------------
'Name: RemoveVocabulary
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This function removes one or more vocabulary items selected in
'the remove list box.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub RemoveVocabulary ()

    '---------------------
    'Variable Declarations
    '---------------------
    'Loop counter.
    Dim i As Integer

    'Full string in the remove box.
    Dim sNameFacet As String

    'Name of the vocabulary and facet.
    Dim sName As String
    Dim sFacet As String
    Dim nFacetId As Integer

    'Start and end for string parsing.
    Dim nStart As Integer
```

```
    Dim nLength As Integer

    'Query to remove the vocabulary.
    Dim sqryRemoveVocabulary As String


    '---------------------------------------
    'All the items selected should be removed
    '---------------------------------------
    For i = 0 To lstNameFacet.ListCount - 1
        If lstNameFacet.Selected(i) = True Then
            sNameFacet = lstNameFacet.List(i)

            'Parse the sNameFacet into name and Facet

            'Format of name is (name)  : (Facet), length of name is position 2 to
            'the character before ).
            nStart = 2
            nLength = InStr(1, sNameFacet, ")") - nStart
            sName = Mid$(sNameFacet, nStart, nLength)

            'From the second ( to just before the last character.
            'Names are unique by the group.
            nStart = InStr(2, sNameFacet, "(") + 1
            nLength = Len(sNameFacet) - nStart
            sFacet = Mid$(sNameFacet, nStart, nLength)

            'Get Group Id
            nFacetId = GetFacetId(sFacet)


            sqryRemoveVocabulary = "DELETE FROM VOCABULARY WHERE VocabularyName = "
            sqryRemoveVocabulary = sqryRemoveVocabulary + "'" + sName + "'" + " AND
FacetId = " + CStr(nFacetId)
            gdbRwf.Execute (sqryRemoveVocabulary)
        End If
    Next


    '-----------------------------------------------------
    'Adjust the classification after the facet is removed
    '-----------------------------------------------------
    AdjustClassification


    '-------------------
    'Refresh boxes again
    '-------------------
    FillBoxes


End Sub

'Global File
'-------------------------------------------------------------------------------
'Name: Main Module
'
'Type: Global Declarations

'Description: All the global variables are declared here.

'Author: Ram Mareddy
'-------------------------------------------------------------------------------

Option Explicit

'-------------------
```

```
'File and path names
'-------------------
Global gsIniFileName As String
Global gsHelpFileName As String
Global gsAppDirectory As String

'----------------
'Public variables
'----------------
Global gsWindowsDirectory As String
Global gsSystemDirectory As String
Global gsTempDirectory As String

'------------------------
'Artifact Physical Details
'------------------------
Global gsArtifactFileName As String
Global gsArtifactFilePath As String
Global gsArtifactAuthor As String
Global gsArtifactDescription As String
Global gnArtifactFileLength As Integer
Global gsClassification As String
Global gsSynonymHits() As String

'---------------
'Login User Name
'---------------
Global gsLoginFirstName As String
Global gsLoginLastName As String

'--------
'Database
'--------
Global gdbRwf As Database
Global Const gnNOT_IN_DATABASE = -9999

'----------------------------------------
'Structure of initialization file variables
'----------------------------------------
'g stands for global - (for the sake of
'consistency, eventhough a structure can
'not be defined in the form).
'udt stands for User Defined Type.
'Ini stands for initialization file.
'This structure can later be expanded.
Type gudtIni
    bMaximized As Integer
    sMainWindow As String
    bToolbar As Integer
    bStatusBar As Integer
    sFontName As String
    bFontBold As Integer
    nFontSize As Single
    sDatabase As String
End Type

'The ini used for this application.
Global gudtRwfIni As gudtIni


'----------------------------
'Spread.VBX related constants
'----------------------------

'function prototypes
Declare Sub SpreadSetText Lib "Spread20.VBX" (SS As Control, ByVal Col As Long, ByVal Row
As Long, lpVar As Variant)
```

```
'All rows, all columns
Global Const SS_ALL_ROWS = -1
Global Const SS_ALL_COLUMNS = -1

'spreadsheet actions
Global Const SS_ACTION_ACTIVE_CELL = 0
Global Const SS_ACTION_GOTO_CELL = 1
Global Const SS_ACTION_SELECT_BLOCK = 2
Global Const SS_ACTION_CLEAR = 3
Global Const SS_ACTION_DELETE_COL = 4
Global Const SS_ACTION_DELETE_ROW = 5
Global Const SS_ACTION_INSERT_COL = 6
Global Const SS_ACTION_INSERT_ROW = 7
Global Const SS_ACTION_LOAD_SPREAD_SHEET = 8
Global Const SS_ACTION_SAVE_ALL = 9
Global Const SS_ACTION_SAVE_VALUES = 10
Global Const SS_ACTION_RECALC = 11
Global Const SS_ACTION_CLEAR_TEXT = 12
Global Const SS_ACTION_PRINT = 13
Global Const SS_ACTION_DESELECT_BLOCK = 14
Global Const SS_ACTION_DSAVE = 15
Global Const SS_ACTION_SET_CELL_BORDER = 16
Global Const SS_ACTION_ADD_MULTISELBLOCK = 17
Global Const SS_ACTION_GET_MULTI_SELECTION = 18
Global Const SS_ACTION_COPY_RANGE = 19
Global Const SS_ACTION_MOVE_RANGE = 20
Global Const SS_ACTION_SWAP_RANGE = 21
Global Const SS_ACTION_CLIPBOARD_COPY = 22
Global Const SS_ACTION_CLIPBOARD_CUT = 23
Global Const SS_ACTION_CLIPBOARD_PASTE = 24
Global Const SS_ACTION_SORT = 25
Global Const SS_ACTION_COMBO_CLEAR = 26
Global Const SS_ACTION_COMBO_REMOVE = 27
Global Const SS_ACTION_RESET = 28
Global Const SS_ACTION_SS_ACTION_SEL_MODE_CLEAR = 29
Global Const SS_ACTION_VMODE_REFRESH = 30
Global Const SS_ACTION_REFRESH_BOUND = 31
Global Const SS_ACTION_SMARTPRINT = 32

'cell type
Global Const SS_CELL_TYPE_DATE = 0
Global Const SS_CELL_TYPE_EDIT = 1
Global Const SS_CELL_TYPE_FLOAT = 2
Global Const SS_CELL_TYPE_INTEGER = 3
Global Const SS_CELL_TYPE_PIC = 4
Global Const SS_CELL_TYPE_STATIC_TEXT = 5
Global Const SS_CELL_TYPE_TIME = 6
Global Const SS_CELL_TYPE_BUTTON = 7
Global Const SS_CELL_TYPE_COMBOBOX = 8
Global Const SS_CELL_TYPE_PICTURE = 9
Global Const SS_CELL_TYPE_CHECKBOX = 10
Global Const SS_CELL_TYPE_OWNER_DRAWN = 11

'sort order
Global Const SS_SORT_ORDER_NONE = 0
Global Const SS_SORT_ORDER_ASCENDING = 1
Global Const SS_SORT_ORDER_DESCENDING = 2

'Sort By
Global Const SS_SORT_BY_ROW = 0
Global Const SS_SORT_BY_COL = 1

'date formats
Global Const SS_CELL_DATE_FORMAT_DDMONYY = 0
Global Const SS_CELL_DATE_FORMAT_DDMMYY = 1
Global Const SS_CELL_DATE_FORMAT_MMDDYY = 2
```

```
Global Const SS_CELL_DATE_FORMAT_YYMMDD = 3

'Static text vertical alignment
Global Const SS_CELL_STATIC_V_ALIGN_BOTTOM = 0
Global Const SS_CELL_STATIC_V_ALIGN_CENTER = 1
Global Const SS_CELL_STATIC_V_ALIGN_TOP = 2

'Time
Global Const SS_CELL_TIME_12_HOUR_CLOCK = 0
Global Const SS_CELL_TIME_24_HOUR_CLOCK = 1

'Unit type
Global Const SS_CELL_UNIT_NORMAL = 0
Global Const SS_CELL_UNIT_VGA = 1
Global Const SS_CELL_UNIT_TWIPS = 2

'horizontal align
Global Const SS_CELL_H_ALIGN_LEFT = 0
Global Const SS_CELL_H_ALIGN_RIGHT = 1
Global Const SS_CELL_H_ALIGN_CENTER = 2

'EditmodeAction
Global Const SS_CELL_EDITMODE_EXIT_NONE = 0
Global Const SS_CELL_EDITMODE_EXIT_UP = 1
Global Const SS_CELL_EDITMODE_EXIT_DOWN = 2
Global Const SS_CELL_EDITMODE_EXIT_LEFT = 3
Global Const SS_CELL_EDITMODE_EXIT_RIGHT = 4
Global Const SS_CELL_EDITMODE_EXIT_NEXT = 5
Global Const SS_CELL_EDITMODE_EXIT_PREVIOUS = 6


'---------------------------
'Visual Basic Constants file
'---------------------------
' Function Parameters
' MsgBox parameters
Global Const MB_OK = 0                  ' OK button only
Global Const MB_OKCANCEL = 1            ' OK and Cancel buttons
Global Const MB_ABORTRETRYIGNORE = 2    ' Abort, Retry, and Ignore buttons
Global Const MB_YESNOCANCEL = 3         ' Yes, No, and Cancel buttons
Global Const MB_YESNO = 4               ' Yes and No buttons
Global Const MB_RETRYCANCEL = 5         ' Retry and Cancel buttons

Global Const MB_ICONSTOP = 16           ' Critical message
Global Const MB_ICONQUESTION = 32       ' Warning query
Global Const MB_ICONEXCLAMATION = 48    ' Warning message
Global Const MB_ICONINFORMATION = 64    ' Information message

Global Const MB_APPLMODAL = 0           ' Application Modal Message Box
Global Const MB_DEFBUTTON1 = 0          ' First button is default
Global Const MB_DEFBUTTON2 = 256        ' Second button is default
Global Const MB_DEFBUTTON3 = 512        ' Third button is default
Global Const MB_SYSTEMMODAL = 4096      'System Modal

' Properties

' Colors
Global Const G_BLACK = 0
Global Const G_BLUE = 1
Global Const G_GREEN = 2
Global Const G_CYAN = 3
Global Const G_RED = 4
Global Const G_MAGENTA = 5
Global Const G_BROWN = 6
Global Const G_LIGHT_GRAY = 7
Global Const G_DARK_GRAY = 8
Global Const G_LIGHT_BLUE = 9
```

```
Global Const G_LIGHT_GREEN = 10
Global Const G_LIGHT_CYAN = 11
Global Const G_LIGHT_RED = 12
Global Const G_LIGHT_MAGENTA = 13
Global Const G_YELLOW = 14
Global Const G_WHITE = 15
Global Const G_AUTOBW = 16

Global Const G_WRONG_COLOR = -1

' WindowState
Global Const NORMAL = 0     ' 0 - Normal
Global Const MINIMIZED = 1 ' 1 - Minimized
Global Const MAXIMIZED = 2 ' 2 - Maximized

' Show parameters
Global Const Modal = 1
Global Const MODELESS = 0

' MsgBox return values
Global Const IDOK = 1                   ' OK button pressed
Global Const IDCANCEL = 2               ' Cancel button pressed
Global Const IDABORT = 3                ' Abort button pressed
Global Const IDRETRY = 4                ' Retry button pressed
Global Const IDIGNORE = 5               ' Ignore button pressed
Global Const IDYES = 6                  ' Yes button pressed
Global Const IDNO = 7                   ' No button pressed


'---------------------------------
'Windows API function declarations
'---------------------------------
Declare Function GetPrivateProfileString Lib "Kernel" (ByVal lpApplicationName As String,
lpKeyName As Any, ByVal lpDefault As String, ByVal lpReturnedString As String, ByVal nSize
As Integer, ByVal lpFileName As String) As Integer
Declare Function GetPrivateProfileInt Lib "Kernel" (ByVal lpApplicationName As String,
ByVal lpKeyName As String, ByVal nDefault As Integer, ByVal lpFileName As String) As
Integer
Declare Function GetWindowsDirectory Lib "Kernel" (ByVal lpBuffer As String, ByVal nSize
As Integer) As Integer
Declare Function GetSystemDirectory Lib "Kernel" (ByVal lpBuffer As String, ByVal nSize As
Integer) As Integer
Declare Function WritePrivateProfileString Lib "Kernel" (ByVal lpApplicationName As
String, lpKeyName As Any, lpString As Any, ByVal lplFileName As String) As Integer

'-------------------------------------------------------------------------------------------
'Name: CenterForm
'
'Type: Subroutine
'
'Parameters: frm is the form that needs to be centered.
'
'Returns: None
'
'Description: Whatever form you feed to this function, it will be centered.
'A simple global implementation of centering the form.
'
'Author : Ram Mareddy
'-------------------------------------------------------------------------------------------
'
Sub CenterForm (frm As Form)


    '-----------
    'Center Form
    '-----------
    frm.Move (screen.Width - frm.Width) / 2, (screen.Height - frm.Height) / 2
```

```
End Sub

'------------------------------------------------------------------------------
'Name: FillFacetsInSearch
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This routine fills the search spread with facets.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------------
'
Sub FillFacetsInInsert ()


    '---------------------
    'Variable Declarations
    '---------------------
    'String that holds the query for the facets.
    Dim sqryFacets As String

    'Dynaset that holds the results.
    Dim dsFacets As dynaset

    'Counter for number of facets.
    Dim nNumFacets As Integer

    'Loop counter.
    Dim i As Integer

    '-----------------------
    'Query for Facet id/name
    '-----------------------
    sqryFacets = "SELECT FacetId, FacetName from FACET"
    Set dsFacets = gdbRwf.CreateDynaset(sqryFacets)

    nNumFacets = 0
    dsFacets.MoveFirst

    Do Until dsFacets.EOF
        nNumFacets = nNumFacets + 1
        dsFacets.MoveNext
    Loop

    dsFacets.MoveFirst
    For i = 1 To nNumFacets
        frmClassification.sprdInsert.Row = i
        frmClassification.sprdInsert.Col = 0
        frmClassification.sprdInsert.Text = dsFacets("FacetName")
        dsFacets.MoveNext
    Next i

End Sub

'------------------------------------------------------------------------------
'Name: FillFacetsInSearch
'
'Type: Subroutine
'
'Parameters: None
'
```

```
'Returns: None
'
'Description: This routine fills the search spread with facets.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------------
'
Sub FillFacetsInSearch ()


    '---------------------
    'Variable Declarations
    '---------------------
    'String that holds the query for the facets.
    Dim sqryFacets As String

    'Dynaset that holds the results.
    Dim dsFacets As dynaset

    'Counter for number of facets.
    Dim nNumFacets As Integer

    'Loop counter.
    Dim i As Integer

    '-----------------------
    'Query for Facet id/name
    '-----------------------
    sqryFacets = "SELECT FacetId, FacetName from FACET"
    Set dsFacets = gdbRwf.CreateDynaset(sqryFacets)

    nNumFacets = 0
    dsFacets.MoveFirst

    Do Until dsFacets.EOF
        nNumFacets = nNumFacets + 1
        dsFacets.MoveNext
    Loop

    dsFacets.MoveFirst
    For i = 1 To nNumFacets
        frmSearch.sprdSearch.Row = i
        frmSearch.sprdSearch.Col = 0
        frmSearch.sprdSearch.Text = dsFacets("FacetName")
        dsFacets.MoveNext
    Next i


End Sub

'--------------------------------------------------------------------------
'Name: FillSpreadCombo
'
'Type: Subroutine
'
'Parameters: lCol is the column number of the cell and lRow is Row number.
'
'Returns: None
'
'Description: This function fills the combobox on the spread, for that
'particular selected cell.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
'
Sub FillInsertCombo (lCol As Long, lRow As Long)
```

```
    '------------
    'Declarations
    '------------
    'Dynasets which store the facet id and the vocabulary.
    Dim dsFacetId As dynaset
    Dim dsVocabulary As dynaset

    'Queries to obtain facet id and vocabulary.
    Dim sqryFacetId As String
    Dim sqryVocabulary As String

    'Current selections of facet and vocabulary on the spread.
    Dim sCurrentVocabulary As String
    Dim sCurrentFacet As String
    Dim nCurrentFacetId As Integer

    'Index of vocabulary in the combo box.
    Dim nVocabularyIndex As Integer

    'Loop counter.
    Dim i As Integer

    'Column number constant.
    Const FACET_COLUMN_NUMBER = 0


    '-------------
    'Get the facet
    '-------------
    frmClassification.sprdInsert.Col = FACET_COLUMN_NUMBER
    frmClassification.sprdInsert.Row = lRow
    sCurrentFacet = frmClassification.sprdInsert.Text


    '----------------------
    'Set the Row and Column
    '----------------------
    frmClassification.sprdInsert.Col = lCol
    frmClassification.sprdInsert.Row = lRow
    sCurrentVocabulary = frmClassification.sprdInsert.Text

    frmClassification.sprdInsert.CellType = SS_CELL_TYPE_COMBOBOX


    '----------------
    'Get the facet id
    '----------------
    sqryFacetId = "SELECT FacetId FROM FACET WHERE FacetName = " + "'" + sCurrentFacet +
"'"
    Set dsFacetId = gdbRwf.CreateDynaset(sqryFacetId)
    nCurrentFacetId = dsFacetId("FacetId")


    '--------------------
    'Get the vocabulary
    '--------------------
    sqryVocabulary = "SELECT VocabularyName FROM VOCABULARY WHERE FacetId = " +
CStr(nCurrentFacetId)
    Set dsVocabulary = gdbRwf.CreateDynaset(sqryVocabulary)


    '---------------------
    'Fill Spread Combo Box
    '---------------------
    'First add ALL.
    frmClassification.sprdInsert.TypeComboBoxIndex = 0
```

```
    frmClassification.sprdInsert.TypeComboBoxString = "ALL"
    If sCurrentVocabulary = "ALL" Then
        nVocabularyIndex = 0
    End If

    i = 1
    Do Until dsVocabulary.EOF
        frmClassification.sprdInsert.TypeComboBoxIndex = i
        frmClassification.sprdInsert.TypeComboBoxString = dsVocabulary("VocabularyName")

        If sCurrentVocabulary = frmClassification.sprdInsert.TypeComboBoxString Then
            nVocabularyIndex = i
        End If

        dsVocabulary.MoveNext
        i = i + 1
    Loop


    '--------------------------------
    'Select the item in the combo box
    '--------------------------------
    frmClassification.sprdInsert.TypeComboBoxCurSel = nVocabularyIndex


End Sub

'------------------------------------------------------------------------------
'Name: FillSpreadCombo
'
'Type: Subroutine
'
'Parameters: lCol is the column number of the cell and lRow is Row number.
'
'Returns: None
'
'Description: This function fills the combobox on the spread, for that
'particular selected cell.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------------
'
Sub FillSearchCombo (lCol As Long, lRow As Long)


    '------------
    'Declarations
    '------------
    'Dynasets which store the facet id and the vocabulary.
    Dim dsFacetId As dynaset
    Dim dsVocabulary As dynaset

    'Queries to obtain facet id and vocabulary.
    Dim sqryFacetId As String
    Dim sqryVocabulary As String

    'Current selections of facet and vocabulary on the spread.
    Dim sCurrentVocabulary As String
    Dim sCurrentFacet As String
    Dim nCurrentFacetId As Integer

    'Index of vocabulary in the combo box.
    Dim nVocabularyIndex As Integer

    'Loop counter.
    Dim i As Integer
```

```
    'Column number constant.
    Const FACET_COLUMN_NUMBER = 0


    '-------------
    'Get the facet
    '-------------
    frmSearch.sprdSearch.Col = FACET_COLUMN_NUMBER
    frmSearch.sprdSearch.Row = lRow
    sCurrentFacet = frmSearch.sprdSearch.Text


    '----------------------
    'Set the Row and Column
    '----------------------
    frmSearch.sprdSearch.Col = lCol
    frmSearch.sprdSearch.Row = lRow
    sCurrentVocabulary = frmSearch.sprdSearch.Text

    frmSearch.sprdSearch.CellType = SS_CELL_TYPE_COMBOBOX


    '----------------
    'Get the facet id
    '----------------
    sqryFacetId = "SELECT FacetId FROM FACET WHERE FacetName = " + "'" + sCurrentFacet +
"'"
    Set dsFacetId = gdbRwf.CreateDynaset(sqryFacetId)
    nCurrentFacetId = dsFacetId("FacetId")


    '--------------------
    'Get the vocabulary
    '--------------------
    sqryVocabulary = "SELECT VocabularyName FROM VOCABULARY WHERE FacetId = " +
CStr(nCurrentFacetId)
    Set dsVocabulary = gdbRwf.CreateDynaset(sqryVocabulary)


    '---------------------
    'Fill Spread Combo Box
    '---------------------
    'First add ALL.
    frmSearch.sprdSearch.TypeComboBoxIndex = 0
    frmSearch.sprdSearch.TypeComboBoxString = "ALL"
    If sCurrentVocabulary = "ALL" Then
        nVocabularyIndex = 0
    End If

    i = 1
    Do Until dsVocabulary.EOF
        frmSearch.sprdSearch.TypeComboBoxIndex = i
        frmSearch.sprdSearch.TypeComboBoxString = dsVocabulary("VocabularyName")

        If sCurrentVocabulary = frmSearch.sprdSearch.TypeComboBoxString Then
            nVocabularyIndex = i
        End If

        dsVocabulary.MoveNext
        i = i + 1
    Loop


    '-------------------------------
    'Select the item in the combo box
    '-------------------------------
    frmSearch.sprdSearch.TypeComboBoxCurSel = nVocabularyIndex
```

```
End Sub

'-------------------------------------------------------------------------
'Name: FillSynonymCombo
'
'Type: Subroutine
'
'Parameters: lCol is the column number of the cell and lRow is Row number.
'
'Returns: None
'
'Description: This function fills the combobox on the spread, for that
'particular selected cell. Items in the combo are Yes, No.
'
'Author: Ram Mareddy
'-------------------------------------------------------------------------
'
Sub FillSynonymCombo (lCol As Long, lRow As Long)


    '---------------------
    'Variable Declarations
    '---------------------
    Dim sCurrentString As String


    '------------------------------
    'Set the cell to combo box type
    '------------------------------
    frmSearch.sprdSearch.Col = lCol
    frmSearch.sprdSearch.Row = lRow
    sCurrentString = frmSearch.sprdSearch.Text

    frmSearch.sprdSearch.CellType = SS_CELL_TYPE_COMBOBOX


    '------------------
    'Fill the combo box
    '------------------
    frmSearch.sprdSearch.TypeComboBoxIndex = 0
    frmSearch.sprdSearch.TypeComboBoxString = "NO"

    frmSearch.sprdSearch.TypeComboBoxIndex = 1
    frmSearch.sprdSearch.TypeComboBoxString = "YES"


    '-------------------------
    'Set the current selection
    '-------------------------
    If sCurrentString = "NO" Then
        frmSearch.sprdSearch.TypeComboBoxCurSel = 0
    Else
        frmSearch.sprdSearch.TypeComboBoxCurSel = 1
    End If


End Sub

'-------------------------------------------------------------------------------------
'Name: GetAppEnvironment
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
```

```
'
'Description: This function gets the environment for the application from the
'initialization file. Name of the initialization file is AppExeName.ini.
'
'Author : Ram Mareddy
'-------------------------------------------------------------------------------
'
Sub GetAppEnvironment ()


    '-----------------------------------------
    'Initialization file has the following format
    '-----------------------------------------
    ' [Settings]
    ' Maximized = 0                    'Window is maximized or minimized.
    ' toolbar = 1                      'Visibility of toolbar.
    ' statusbar = 1                    'Visibility of statusbar.
    ' mainwindow=870 645 10290 5880    'If window is maximized, what are the co-ordinates?
    ' Fontname = NewBrunswick          'What font?
    ' Fontbold = 0                     'Whether the font is bold or not?
    ' FontSize = 8                     'Font size.

    ' [DataSource]
    ' Default=c:\rwf\rwf.mdb


    '---------------------
    'Variable Declarations
    '---------------------
    'Section in the ini file.
    Dim sSection As String

    'Entry in that particular section.
    Dim sEntry As String * 20

    'Default string value for the entry.
    Dim sDefault As String

    'Default integer value for the entry.
    Dim nDefault As Integer

    'Return value from the ini file.
    Dim sReturnBuffer As String

    'Length of the string you are giving.
    Dim nReturnBuffer As Integer

    'Initialization file name.
    Dim sFileName As String

    'Number of bytes returned.
    Dim nCount As Integer

    'Message in the message box.
    Dim sMessage As String

    'Title for the message box.
    Dim sTitle As String

    'Response from the message box.
    Dim nResponse As Integer


    '----------------------
    'App name and directory
    '----------------------
    'First let us get the name of this application, from which we can get names of ini
```

```
'and hlp files. (We know the name of this file, but just to be generic).
'Get the path and executable name and then add ini to it.
gsAppDirectory = App.Path + "\"
gsIniFileName = gsAppDirectory + App.EXEName + ".ini"

'Also the name of the help file.
gsHelpFileName = gsAppDirectory + App.EXEName + ".hlp"


'------------------
'Get the Window mode
'------------------
sSection = "Settings"
sEntry = "Maximized"
nDefault = 1     'If Maximized is not there, open the window full screen.
sFileName = gsIniFileName

nCount = GetPrivateProfileInt(sSection, sEntry, nDefault, sFileName)

If nCount = 1 Then
    gudtRwfIni.bMaximized = True
Else
    gudtRwfIni.bMaximized = False
End If


'----------------------
'Get the Toolbar status
'----------------------
sSection = "Settings"
sEntry = "Toolbar"
nDefault = 1     'If Toolbar is not there, put the toolbar.
sFileName = gsIniFileName

nCount = GetPrivateProfileInt(sSection, sEntry, nDefault, sFileName)

If nCount = 1 Then
    gudtRwfIni.bToolbar = True
Else
    gudtRwfIni.bToolbar = False
End If


'------------------------
'Get the Statusbar status
'------------------------
sSection = "Settings"
sEntry = "Statusbar"
nDefault = 1     'If Statusbar is not there, put the statusbar.
sFileName = gsIniFileName

nCount = GetPrivateProfileInt(sSection, sEntry, nDefault, sFileName)

If nCount = 1 Then
    gudtRwfIni.bStatusBar = True
Else
    gudtRwfIni.bStatusBar = False
End If

'--------------------------------
'Get the main window co-ordinates
'--------------------------------
sSection = "Settings"
sEntry = "MainWindow"
sDefault = "1065 1065 7020 5130"     'This is like center of the screen.
sReturnBuffer = Space$(81)
nReturnBuffer = Len(sReturnBuffer)
```

```
    sFileName = gsIniFileName

    nCount = GetPrivateProfileString(sSection, ByVal sEntry, sDefault, sReturnBuffer,
nReturnBuffer, sFileName)

    gudtRwfIni.sMainWindow = Left$(sReturnBuffer, nCount)

    '-----------------
    'Get the font name
    '-----------------
    sSection = "Settings"
    sEntry = "FontName"
    sDefault = "MS Sanserif"
    sReturnBuffer = Space$(81)
    nReturnBuffer = Len(sReturnBuffer)
    sFileName = gsIniFileName

    nCount = GetPrivateProfileString(sSection, ByVal sEntry, sDefault, sReturnBuffer,
nReturnBuffer, sFileName)

    gudtRwfIni.sFontName = Left$(sReturnBuffer, nCount)


    '--------------------
    'Get font bold status
    '--------------------
    sSection = "Settings"
    sEntry = "FontBold"
    nDefault = 0     'If bold status is omitted, do not use bold.
    sFileName = gsIniFileName

    nCount = GetPrivateProfileInt(sSection, sEntry, nDefault, sFileName)

    If nCount = 1 Then
        gudtRwfIni.bFontBold = True
    Else
        gudtRwfIni.bFontBold = False
    End If


    '-------------
    'Get font size
    '-------------
    sSection = "Settings"
    sEntry = "FontSize"
    nDefault = 8     'If font size is omitted, use size 8.
    sFileName = gsIniFileName

    nCount = GetPrivateProfileInt(sSection, sEntry, nDefault, sFileName)

    gudtRwfIni.nFontSize = nCount


    '--------------------------
    'Get the default datasource
    '--------------------------
    sSection = "DataSource"
    sEntry = "Default"
    sDefault = ""                        'When there is no database specified, we are in
trouble. We have to warn the user and quit.
    sReturnBuffer = Space$(81)
    nReturnBuffer = Len(sReturnBuffer)
    sFileName = gsIniFileName

    nCount = GetPrivateProfileString(sSection, ByVal sEntry, sDefault, sReturnBuffer,
nReturnBuffer, sFileName)
```

```
    gudtRwfIni.sDatabase = Left$(sReturnBuffer, nCount)

    'If database is not specified...
    sMessage = "The default database file is not specified. "
    sMessage = sMessage + "Under [DataSource] section in the initialization file add
Default = <Database Name>. "
    sMessage = sMessage + "Initialization file name is " + gsIniFileName + "."
    sTitle = "Initialization Error"
    If gudtRwfIni.sDatabase = "" Then
        nResponse = MsgBox(sMessage, MB_OK + MB_ICONSTOP, sTitle)
        End
    End If


End Sub

'--------------------------------------------------------------------------
'Name: GetFacetId
'
'Type: Subroutine
'
'Parameters: sFacetName
'
'Returns: FacetId
'
'Description: The Id from database is returned for the given facet name.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
'
Function GetFacetId (sFacetName As String) As Integer

    '---------------------
    'Variable Declarations
    '---------------------
    'String that holds the query for the facets.
    Dim sqryFacetId As String

    'Dynaset that holds the results.
    Dim dsFacetId As dynaset


    '-----------------------
    'Query for Facet id/name
    '-----------------------
    sqryFacetId = "SELECT FacetId from FACET WHERE FacetName = " + "'" + sFacetName + "'"
    Set dsFacetId = gdbRwf.CreateDynaset(sqryFacetId)

    GetFacetId = dsFacetId("FacetId")


End Function

'--------------------------------------------------------------------------
'Name: GetNewClassification
'
'Type: Subroutine
'
'Parameters: sClassification, sSynonymId, nPosition
'
'Returns: None
'
'Description: The classification for a synonym is obtained with the help of
'old classification and the position of the synonym and its id.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
```

```
'
Function GetNewClassification (sClassification As String, sSynonymId As String, nPosition
As Integer) As String

    '------------
    'Declarations
    '------------
    Dim nNumCommas As Integer
    Dim nCounter As Integer
    Dim nStart As Integer
    Dim nEnd As Integer
    Dim nStartFlag As Integer
    Dim nEndFlag As Integer
    Dim sNewClassification As String


    '--------------------------------------
    'Get the number of synonyms in the string
    '--------------------------------------
    'Count the number of commas in the function.
    nNumCommas = 0
    nStartFlag = False
    nEndFlag = False
    For nCounter = 1 To Len(sClassification)

        If StrComp(Mid$(sClassification, nCounter, 1), ",") = 0 Then
            nNumCommas = nNumCommas + 1
        End If

        If (nPosition = nNumCommas + 1) And (nStartFlag = False) Then
            nStart = nCounter + 1
            nStartFlag = True
        End If

        If (nPosition = nNumCommas) And (nEndFlag = False) Then
            nEnd = nCounter - 1
            nEndFlag = True
        End If


    Next


    '---------------
    'Adust the string
    '---------------
    sNewClassification = Left$(sClassification, nStart - 1)
    sNewClassification = sNewClassification + sSynonymId
    sNewClassification = sNewClassification + Right$(sClassification, Len(sClassification)
- nEnd)

    GetNewClassification = sNewClassification


End Function

'-------------------------------------------------------------------------------
'Name: GetNthString
'
'Type: Function
'
'Parameters: sString is the string to be separated, nN is the number of string
'segment, sSeparator is the separator string.
'
'Retunrs: Nth string segment of the parent string.
'
'Description: This function returns part of the string which is divided by
```

```
'separator, like space, comma, etc. For example GetNthString ("21 54 36 67", 2, " ")
'will return "54".
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------------
'
Function GetNthString (sString As String, nN As Integer, sSeparator As String) As String

    '---------------------
    'Variable Declarations
    '---------------------
    'Starting position of the string.
    Dim nStart As Integer

    'Ending position of the string.
    Dim nEnd As Integer

    'Counter variable.
    Dim i As Integer


    '--------------------
    'Start and End points
    '--------------------
    'Start position.
    nStart = 1
    For i = 1 To nN - 1
        nStart = InStr(nStart, sString, sSeparator) + 1
    Next

    'For the end position of the string, just get the next space.
    nEnd = InStr(nStart, sString, sSeparator)


    '----------
    'Nth string
    '----------
    'If the separator is not found, then 0 is returned. It could be the last string part.
    If nEnd = 0 Then
        nEnd = Len(sString) + 1
    End If

    'The string of our interest is in between nStart and nEnd.
    GetNthString = Mid$(sString, nStart, nEnd - nStart)

End Function

'--------------------------------------------------------------------------------
'Name: GetNumFacets
'
'Type: Function
'
'Parameters: None
'
'Returns: Number of facets
'
'Description: This routine gets the facets declared in the system and
'returns the number.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------------
'
Function GetNumFacets () As Integer


    '---------------------
    'Variable Declarations
```

```
    '--------------------
    'String that holds the query for the facets.
    Dim sqryFacets As String

    'Dynaset that holds the results.
    Dim dsFacets As dynaset

    'Counter for number of facets.
    Dim nNumFacets As Integer


    '-----------------------
    'Query for Facet id/name
    '-----------------------
    sqryFacets = "SELECT FacetId, FacetName from FACET"
    Set dsFacets = gdbRwf.CreateDynaset(sqryFacets)

    nNumFacets = 0
    dsFacets.MoveFirst

    Do Until dsFacets.EOF
        nNumFacets = nNumFacets + 1
        dsFacets.MoveNext
    Loop

    GetNumFacets = nNumFacets

End Function

'------------------------------------------------------------------------------
'Name: GetNumSynonyms
'
'Type: Function
'
'Parameters: sSynonymString
'
'Returns: Number of synonyms
'
'Description: From the synonym string, number of synonyms for that facet are
'returned.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------------
'
Function GetNumSynonyms (sSynonymString As String) As Integer

    '------------
    'Declarations
    '------------
    Dim nNumCommas As Integer
    Dim nCounter As Integer


    '--------------------------------------
    'Get the number of synonyms in the string
    '--------------------------------------
    'Count the number of commas in the function.
    nNumCommas = 0
    For nCounter = 1 To Len(sSynonymString)
        If Mid$(sSynonymString, nCounter, 1) = "," Then
            nNumCommas = nNumCommas + 1
        End If
    Next nCounter

    'Number of synonyms is number of commas plus one.
    GetNumSynonyms = nNumCommas + 1
```

```
End Function

'-------------------------------------------------------------------------
'Name: GetPublicEnvironment
'
'Type: Subroutine
'
'Parameters: psWindowsDirectory that stores the directory where Microsoft
'Windows is installed, psSystemDirectory that stores the system directory
'of Microsoft Windows, psTempDirectory that stores the temporary directory
'of Microsoft Windows.
'
'Returns: None
'
'Description: This function gets the windows, system, and temporary
'directories. Windows directory is where the ini file is kept and system
'directory is where the dlls are kept.
'
'Author: Ram Mareddy
'-------------------------------------------------------------------------
'
Sub GetPublicEnvironment (psWindowsDirectory As String, psSystemDirectory As String,
psTempDirectory As String)


    '--------------------
    'Variable Declarations
    '--------------------
    'Holding string for directory.
    Dim sDirectory As String

    'Size of the directory string.
    Dim nDirectory As Integer

    'Return value.
    Dim nCount As Integer


    '-------------------------
    'Get the Windows directory
    '-------------------------
    sDirectory = Space$(81)
    nDirectory = Len(sDirectory)
    nCount = GetWindowsDirectory%(sDirectory, nDirectory)
    psWindowsDirectory = Left$(sDirectory, nCount)
    'Add a backslash ("\") to the end, if it is not there.
    If Mid$(psWindowsDirectory, Len(psWindowsDirectory), 1) <> "\" Then
        psWindowsDirectory = psWindowsDirectory + "\"
    End If


    '-----------------------
    'Get the System directory
    '-----------------------
    sDirectory = Space$(81)
    nDirectory = Len(sDirectory)
    nCount = GetSystemDirectory%(sDirectory, nDirectory)
    psSystemDirectory = Left$(sDirectory, nCount)
    'Add a backslash ("\") to the end, if it is not there.
    If Mid$(psSystemDirectory, Len(psSystemDirectory), 1) <> "\" Then
        psSystemDirectory = psSystemDirectory + "\"
    End If


    '---------------------------
    'Get the Temporary directory
    '---------------------------
```

```
    'Variable Declarations
    '---------------------
    'String that holds the query for the vocabulary id.
    Dim sqryVocabularyId As String

    'Dynaset that holds the results.
    Dim dsVocabularyId As dynaset


    '-----------------------
    'Query for vocabulary id
    '-----------------------
    'If the vocabulary name is ALL, then it has an id 0.
    If sVocabularyName = "ALL" Then
        GetVocabularyId = 0

    Else
        sqryVocabularyId = "SELECT VocabularyId FROM VOCABULARY WHERE VocabularyName = " +
"'" + sVocabularyName + "'"
        Set dsVocabularyId = gdbRwf.CreateDynaset(sqryVocabularyId)

        dsVocabularyId.MoveFirst

        GetVocabularyId = dsVocabularyId("VocabularyId")

    End If

End Function

'-------------------------------------------------------------------------------
'Name: IsArtifactInThisClass
'
'Type: Function
'
'Parameters: sClassification is the input classification, sCurrentArtifact is
'the classification of the retrieved artifact.
'
'Returns: True or False according to the artifact belonging to this class.
'
'Description: This routine takes two strings and compares them for equality.
'A 0 in the first string can be substituted for any character in the second
'string.
'
'Author: Ram Mareddy
'-------------------------------------------------------------------------------
'
Function IsArtifactInThisClass (sClassification As String, sCurrentArtifact As String) As
Integer


    '---------------------
    'Variable Declarations
    '---------------------
    Dim nFirstStart As Integer
    Dim nFirstEnd As Integer
    Dim sFirstString As String

    Dim nSecondStart As Integer
    Dim nSecondEnd As Integer
    Dim sSecondString As String


    '-------------------------------------
    'If both are identical then return true
    '-------------------------------------
    If sClassification = sCurrentArtifact Then
        IsArtifactInThisClass = True
```

```
    Else

        nFirstStart = 1
        nFirstEnd = 1
        nSecondStart = 1
        nSecondEnd = 1
        IsArtifactInThisClass = True

        While (nFirstStart < Len(sClassification))
            'Get the first string segment.
            nFirstEnd = InStr(nFirstStart, sClassification, ",")
            sFirstString = Mid$(sClassification, nFirstStart, nFirstEnd - nFirstStart)

            'Get the second string segment.
            nSecondEnd = InStr(nSecondStart, sCurrentArtifact, ",")
            sSecondString = Mid$(sCurrentArtifact, nSecondStart, nSecondEnd - nFirstStart)

            'If these the first string or second string has 0, then it is OK.
            'Otherwise these strings should be identical.
            If (sFirstString <> "0") And (sSecondString <> "0") Then
                If sFirstString <> sSecondString Then
                    IsArtifactInThisClass = False
                End If
            End If

            nFirstStart = nFirstEnd + 1
            nSecondStart = nSecondEnd + 1

        Wend

    End If

End Function

'--------------------------------------------------------------------------
'Name: IsSynonymVocab
'
'Type: Function
'
'Parameters: sSynonym, sFacet
'
'Returns: True/False
'
'Description: Synonym might be in the thesaurus, but is it part of the
'vocabulary for that particular facet.
'
'Author: Ram Mareddy
'--------------------------------------------------------------------------
'
Function IsSynonymVocab (sSynonym, sFacet)


    '------------
    'Declarations
    '------------
    Dim sqryVocabulary As String
    Dim dsVocabulary As dynaset


    '-----------------
    'Search vocabulary
    '-----------------
    sqryVocabulary = "SELECT * FROM Vocabulary where VocabularyName = '" + sSynonym + "'"
    Set dsVocabulary = gdbRwf.CreateDynaset(sqryVocabulary)

    If dsVocabulary.RecordCount > 0 Then
```

```
        IsSynonymVocab = dsVocabulary("VocabularyId")
    Else
        IsSynonymVocab = gnNOT_IN_DATABASE
    End If

    dsVocabulary.Close

End Function

Sub Login ()

    frmLogin.Show Modal

End Sub

'------------------------------------------------------------------------
'Name: Main
'
'Type: Subroutine
'
'Parameters: None
'
'Returns: None
'
'Description: This is the start-up module. So, all the initialization and
'start-up code is placed here.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Sub Main ()

    'Read the global information.
    GetPublicEnvironment gsWindowsDirectory, gsSystemDirectory, gsTempDirectory

    'Read the initialization file.
    GetAppEnvironment

    'Open the database - if there are any problems, we will just quit.
    OpenDb gudtRwfIni.sDatabase

    'Login is for the purpose of metrics rather than security.
    Login

    'Now show the main form.
    frmmain.Show

End Sub

'------------------------------------------------------------------------
'Name: OpenDb
'
'Type: Subroutine
'
'Parameters: psDatabase is the name of the database to open.
'
'Returns: None
'
'Description: Try to open the database that is supplied as the parameter.
'If there are any errors, inform the user and quit the application, because it's
'an absolute must that the database be there.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
```

```
Sub OpenDb (psDatabase As String)

    '--------------------
    'Variable Declaration
    '--------------------
    'Response from the message box.
    Dim nResponse As Integer

    'Message in the message box.
    Dim sMessage

    'Title of the message box.
    Dim sTitle As String


    '------------
    'Error prompt
    '------------
    On Error Resume Next


    '--------------------
    'Opening the database
    '--------------------
    Set gdbRwf = OpenDatabase(psDatabase)

    If Err Then
        sMessage = "An error has occured in opening the database for Reuse With Facets."
        sMessage = sMessage + "Database name you specified in the ini file (" +
gsIniFileName + ") - " + gudtRwfIni.sDatabase + " - " + "does not  exist."
        sTitle = "Open Database Error"
        nResponse = MsgBox(sMessage, MB_OK + MB_ICONSTOP, sTitle)
        End
    End If

End Sub

'------------------------------------------------------------------------
'Name: SearchClassification
'
'Type: Function
'
'Parameters: sClassification is the classification string.
'
'Returns: Number of Hits
'
'Description: This routine gets the number of artifacts in the system with
'given classification.
'
'Author: Ram Mareddy
'------------------------------------------------------------------------
'
Function SearchRepWithOutSynonyms (sClassification As String) As Integer

    '--------------------
    'Variable Declarations
    '--------------------
    'String that holds the query for the vocabulary id.
    Dim sqryClassification As String

    'Dynaset that holds the results.
    Dim dsClassification As dynaset

    'Number of hits for this classification.
    Dim nNumHits As Integer

    'Classification string of the current artifact.
```

```
    Dim sCurrentArtifact As String


    '------------------------
    'Query for classification
    '------------------------
    sqryClassification = "SELECT VocabularyIdString FROM ARTIFACT"
    Set dsClassification = gdbRwf.CreateDynaset(sqryClassification)

    nNumHits = 0
    Do Until dsClassification.EOF
        sCurrentArtifact = dsClassification("VocabularyIdString")
        If IsArtifactInThisClass(sClassification, sCurrentArtifact) = True Then
            nNumHits = nNumHits + 1
        End If

        dsClassification.MoveNext
    Loop

    SearchRepWithOutSynonyms = nNumHits


End Function

'-------------------------------------------------------------------------------
'Name: SearchRepWithSynonyms
'
'Type: Function
'
'Parameters: sClassification, sSynonymFlagString
'
'Returns: None
'
'Description: Repository is searched for the artifacts matching the current
'classification, using synonyms.
'
'Author: Ram Mareddy
'-------------------------------------------------------------------------------
'
Function SearchRepWithSynonyms (sClassification As String, sSynonymFlagString As String)
As Integer

    '---------------------
    'Variable Declarations
    '---------------------
    'String that holds the query for the vocabulary id.
    Dim sqryClassification As String

    'Dynaset that holds the results.
    Dim dsClassification As dynaset

    'Number of hits for this classification.
    Dim nNumHits As Integer

    'Classification string of the current artifact.
    Dim sCurrentArtifact As String
    Dim i As Integer
    Dim sSynonymId As String
    Dim sVocabulary As String
    Dim sFacet As String
    Dim nNumSynonyms As Integer
    Dim sqryThesaurus As String
    Dim dsThesaurus As dynaset

    'Synonym related.
    Dim sSynonymString As String
    Dim nSynonymCounter As Integer
```

```
    Dim sSynonymName As String
    Dim sNewClassification As String


    '---------------
    'Make synonym set
    '---------------
    For i = 1 To frmSearch.sprdSearch.MaxRows
        sSynonymId = GetNthString(sSynonymFlagString, i, ",")

        'If synonym is to be used...
        If sSynonymId = "1" Then
            frmSearch.sprdSearch.Row = i
            frmSearch.sprdSearch.Col = 1
            sVocabulary = frmSearch.sprdSearch.Text

            frmSearch.sprdSearch.Row = i
            frmSearch.sprdSearch.Col = 0
            sFacet = frmSearch.sprdSearch.Text

            'Search in the thesaurus for this word.
            sqryThesaurus = "SELECT Word, SynonymString FROM THESAURUS WHERE Word = " +
"'" + sVocabulary + "'"
            Set dsThesaurus = gdbRwf.CreateDynaset(sqryThesaurus)

            'See if there is anything in that.
            If dsThesaurus.RecordCount > 0 Then

                'Get the synonym string.
                sSynonymString = dsThesaurus("SynonymString")

                'Get the number of synonyms. This is number of commas plus 1
                'in the string.
                nNumSynonyms = GetNumSynonyms(sSynonymString)

                'Parse it and get id for each of the synonyms.
                For nSynonymCounter = 1 To nNumSynonyms
                    sSynonymName = GetNthString(sSynonymString, nSynonymCounter, ",")

                    'Check to see if this word is there in the vocabulary.
                    sSynonymId = IsSynonymVocab(sSynonymName, sFacet)

                    If sSynonymId <> Str$(gnNOT_IN_DATABASE) Then
                        'With this new id change the classification string.
                        'Give the new id and position in the string.
                        sNewClassification = GetNewClassification(sClassification,
sSynonymId, i)


                        'See if this artifact is in the database.
                        sqryClassification = "SELECT VocabularyIdString FROM ARTIFACT"
                        Set dsClassification = gdbRwf.CreateDynaset(sqryClassification)

                        Do Until dsClassification.EOF
                            sCurrentArtifact = dsClassification("VocabularyIdString")
                            If IsArtifactInThisClass(sNewClassification, sCurrentArtifact)
= True Then

                                'Store this in a global string for later retrieval.
                                nNumHits = nNumHits + 1
                                ReDim Preserve gsSynonymHits(nNumHits)
                                gsSynonymHits(nNumHits) = sCurrentArtifact

                            End If

                            dsClassification.MoveNext
                        Loop
                    End If
```

```
                Next
            End If
        End If
    Next i


    'dsClassification.Close
    'dsThesaurus.Close
    SearchRepWithSynonyms = nNumHits

End Function

'---------------------------------------------------------------------------
'Name: SetMainWindow
'
'Type: Subroutine
'
'Parameters: frm is the form class to be set.
'
'Returns: None
'
'Description: This function sets the main window of the application
'according to the values obtained from the initialization file.
'
'Author: Ram Mareddy
'---------------------------------------------------------------------------
'
Sub SetMainWindow (frm As Form)

    '-----------------------
    'Set window co-ordinates
    '-----------------------
    If gudtRwfIni.bMaximized = True Then
        frm.WindowState = MAXIMIZED
    Else
        'Window co-ordinates are saved in gsMainWindow. And we need to separate them.
        frm.Left = Val(GetNthString(gudtRwfIni.sMainWindow, 1, " "))
        frm.Top = Val(GetNthString(gudtRwfIni.sMainWindow, 2, " "))
        frm.Width = Val(GetNthString(gudtRwfIni.sMainWindow, 3, " "))
        frm.Height = Val(GetNthString(gudtRwfIni.sMainWindow, 4, " "))
    End If


End Sub
```

VITA

Ram R. Mareddy

Candidate for the Degree of

Master of Science

Thesis: AN IMPLEMENTATION OF THE FACETED CLASSIFICATION SYSTEM FOR SOFTWARE REUSE

Major Field: Computer Science

Biographical:

Personal Data: Born in Eatoor, A.P., India, September 19, 1967, son of Vasantha and Somi Reddy.

Education: Received Bachelor of Technology degree in Mechanical Engineering from Jawaharlal Nehru Technological University in May 1988. Completed the requirements for the Master of Science degree at Oklahoma State University in December 1994.

Professional Experience: Management Trainee, Steel Authority of India Ltd., India May 1988 - April 1989; Engineer, Services and Trade Company, Sultanate of Oman, May 1989 - December 1989; Research Assistant, Department of Psychology, Oklahoma State University, August 1991 - November 1992; Programmer, A La Mode, Inc., Edmond, OK, May 1992 - August 1992; PC Programmer, Mirage Resorts, Inc., Las Vegas, NV, December 1992 - June 1994; Programmer/Analyst, Roadway Logistics, Akron, OH, August 1994 - Present.