Unified Medical Language System Knowledge Source Server (UMLSKS)

User's Guide

Version 4.2

July, 2004

Revision 1.0

Table of Contents

About This Guide vii
Audience vii
Release Notes vii
How to Use This Guide vii
If You Need Help… vii

1 Introduction 1-1
1.1 Requirements Summary 1-1
1.1.1 Key Requirements and Design Goals 1-1
1.2 External Interfaces 1-1
1.3 Performance, Sizing and Loading 1-2
1.4 Logical Architecture 1-2
1.5 Physical Architecture 1-2

2 Using the UMLSKS Web Interface 2-1
2.1 Obtaining Access to the UMLSKS 2-1
2.2 Logging into the UMLSKS 2-1
2.3 Navigating the UMLSKS Interface 2-2
2.3.1 Downloading UMLS Knowledge Sources 2-3
2.3.2 Downloading UMLSKS Application Programmer Interface (API) 2-4
2.3.3 Viewing UMLSKS Documentation 2-4
2.3.4 Accessing UMLSKS Resources 2-5
2.3.5 Performing Quick Searches 2-6
2.3.6 Advanced Searching of UMLS Knowledge Sources 2-7
2.4 Accessing the Metathesaurus 2-8
2.4.1 Basic Searching 2-9
2.4.2 Advanced Searching 2-9
2.4.3 Viewing Concept Details 2-13
2.4.4 Viewing Term Details 2-15
2.4.5 Viewing Raw UMLS ASCII Records 2-16
2.5 Viewing the Semantic Network 2-19
2.5.1 Viewing a Semantic Type 2-21
2.5.2 Viewing a Semantic Relation 2-28
2.6 Using the SPECIALIST Lexicon 2-34
2.7 Updating Your View/Profile 2-35
2.8 Logging out of the UMLSKS 2-36

3 Using the UMLSKS Command-line Interface 3-1
3.1 Standard Options 3-1
3.1.1 Controlling Command-Line Input 3-1
3.1.2 Controlling Command-Line Output 3-2
3.1.3 Restricting Search to a Set of Sources 3-2
3.2 Requesting Concept Details 3-2

3.2.1 Direct Concept-Related Options 3-2
3.2.2 Concept's Relation Options 3-2
3.2.3 Associated Expressions Option 3-3
3.2.4 Co-occurring Concepts Option 3-3
3.2.5 Locator Option 3-4
3.3 Requesting Term Details 3-4
3.3.1 Synonyms/Term Variants Option 3-4
3.3.2 MeSH Entry Terms Option 3-4
3.3.3 Allowable Topical Qualifiers Option 3-4
3.3.4 Qualified Terms Option 3-5
3.3.5 String Attributes Option 3-5
3.4 Viewing Raw Metathesaurus Database Tables 3-6
3.5 Requesting the UMLS Release Year 3-6

4 Using the XML Query Facility 4-1
4.1 Retrieving Metathesaurus, Semantic Network, and SPECIALIST Lexicon Data using an XML Query 4-1
4.2 Structure of an XML Query 4-2
4.3 Metathesaurus and Semantic Network Table Contents 4-4
4.3.1 Table ‘mrcon’ 4-4
4.3.2 Table ‘mrso’ 4-4
4.3.3 Table ‘mrdef’ 4-4
4.3.4 Table ‘mrsty’ 4-4
4.3.5 Table ‘mrrel’ 4-5
4.3.6 Table ‘mratx’ 4-5
4.3.7 Table ‘mrcxt’ 4-5
4.3.8 Table ‘mrcoc’ 4-7
4.3.9 Table ‘mrsat’ 4-7
4.3.10 Table ‘mrlo’ 4-8
4.3.11 Table ‘mrxns_eng’ 4-9
4.3.12 Table ‘mrxnw_eng’ 4-9
4.3.13 Table ‘mrxw_eng’ 4-9
4.3.14 Table ‘mrcui’ 4-9
4.3.15 Table ‘mrsab’ 4-10
4.3.16 Table ‘mshqual’ 4-11
4.3.17 Table ‘strattr’ 4-11
4.3.18 Table ‘ttys’ 4-11
4.3.19 Table ‘cots’ 4-11
4.3.20 Table ‘stts’ 4-11
4.3.21 Table ‘rels’ 4-11
4.3.22 Table ‘srdef’ 4-12
4.3.23 Table ‘srgrp’ 4-12
4.3.24 Table ‘srstr’ 4-12
4.3.25 Table ‘srstre1’ 4-13
4.3.26 Table ‘srstre2’ 4-13
4.3.27 Table ‘mrconso’ 4-13
4.3.28 Table ‘mrdoc’ 4-13

4.3.29 Table ‘mrhier’ 4-14
4.3.30 Table ‘mrhist’ 4-14
4.3.31 Table ‘mrmap’ 4-14
4.4 XML Query Example 4-15

Acronym List

API	Application Programmer Interface
CUI	Concept Unique Identifier
FTP	File Transfer Protocol
HHS	Health and Human Services
HTTP	Hypertext Transfer Protocol
UMLSKS	Knowledge Source Server
NIH	National Institutes of Health
NLM	National Library of Medicine
RAM	Random Access Memory
RMI	Remote Method Invocation
UMLS	Unified Medical Language System
XML	eXtensible Markup Language

List of Figures

Figure 1 - Logical System Diagram 1-2
Figure 2 - Physical System Environment 1-3
Figure 3 – UMLSKS Request Account Area 2-1
Figure 4 – UMLSKS Login Area 2-2
Figure 5 – UMLSKS Home Page 2-3
Figure 6 – Download Topic Menu 2-4
Figure 7 – Accessing UMLSKS Documentation 2-5
Figure 8 – Accessing UMLSKS Resources 2-6
Figure 9 – Performing Quick Searches 2-7
Figure 10 – Advanced Searching of UMLS Knowledge Sources 2-8
Figure 11 – Basic Searching of the Metathesaurus 2-9
Figure 12 – Options for Metathesaurus Advanced Searching 2-10
Figure 13 – Metathesaurus Focused Searching 2-11
Figure 14 – Metathesaurus Concept List 2-14
Figure 15 – Metathesaurus Concept Display 2-15
Figure 16 – Metathesaurus Term Display 2-16
Figure 17 – Accessing Metathesaurus Raw ASCII Records 2-17
Figure 18 – Metathesaurus UMLS ASCII Records 2-18
Figure 19 – Semantic Network Browser 2-19
Figure 20 – Viewing a Semantic Type 2-21
Figure 21 – Browsing Semantic Types 2-22
Figure 22 – Show Concepts with a Semantic Type 2-23
Figure 23 – Concepts with a Semantic Type 2-24
Figure 24 – Show Semantic Relations for a Semantic Type 2-25
Figure 25 – Semantic Relations for a Semantic Type 2-26
Figure 26 – Show ASCII Relational Records for a Semantic Type 2-27
Figure 27 – ASCII Relational Records for a Semantic Type 2-28
Figure 28 – Viewing a Semantic Relation 2-29
Figure 29 – Browsing Semantic Relations 2-30
Figure 30 – Show Semantic Types that Participate in a Semantic Relation 2-31
Figure 31 – Semantic Types for a Semantic Relation 2-32
Figure 32 – Show ASCII Relational Records for a Semantic Relation 2-33
Figure 33 – ASCII Relational Records for a Semantic Type 2-33
Figure 34 – Accessing the SPECIALIST Lexicon 2-34
Figure 35 – Results from the SPECIALIST Lexicon 2-35
Figure 36 – Updating Your User Profile 2-36
Figure 37 – Logging out of the UMLSKS 2-36
Figure 38 – Requesting XML Query Execution 4-2

Revision History

September 17,2002	Initial revision from Version 2.1
July 2003	Addition of 4.0 methods, including `getLexicalRecords, listMetaTableNames, listSemNetTableNames, findBasicSemType, findBasicSemRelation, getBasicAssociativeRelations.`
November 2003	Addition of `KSSRetrieverV4_0` interface components.
April 2004	Separated into two documents: User's Guide for the web interface and a "Developer's Guide" containing information an application system developer would need to integrate with the UMLSKS.
July 2004	Updated to include information about new Version 4.2 API commands and modified web interface components.

About This Guide

This guide describes the installation, usage, and programmatic interface for the UMLS Knowledge Source Server (UMLSKS).

Audience

This guide was developed to aid users in navigating the pages and functions of the UMLSKS web interface.

Release Notes

Please refer to the Release Bulletin found in the umlsks.nlm.nih.gov web site for detailed information about the changes found in each release of the UMLSKS software.

How to Use This Guide

This manual contains the following chapters:

- *Chapter 1 - Introduction* describes the basic features and architecture of the UMLSKS.
- *Chapter 2 -Using the UMLSKS Web Interface* provides users with detailed instructions on interfacing with the UMLSKS using a standard web browser (Netscape, Internet Explorer, etc.).
- *Chapter 3 - Using the UMLSKS Command-line Interface* provides users with detailed instructions on interfacing with the UMLSKS using a web browser and a set of text commands.
- *Chapter 4 - Using the XML Query Facility* describes how to use the querying facility of the UMLSKS wherein users build XML queries to be executed.

If You Need Help…

If you run into problems with the UMLSKS, please first consult the User's Guide for detailed instructions on usage and known problems with the system. Should you still encounter problems, please send a detailed message via email to the development team at the following email address: `umlsks@nlm.nih.gov`. Be sure to include the release version number for the UMLSKS system with your mailing.

1 Introduction

The Unified Medical Language System (UMLS) Knowledge Sources and related lexical programs, developed at the U.S. National Library of Medicine (NLM), provide access to the UMLS. The Metathesaurus, the Semantic Network, and the SPECIALIST Lexicon are part of the UMLS and are designed primarily for use by system developers. They are meant to be consulted and used by applications programs to interpret and refine user queries, to map the user's terms to appropriate controlled vocabularies and classification schemes, to interpret natural language, and to assist in structured data creation. They are also useful as reference tools for database builders, librarians and other information professionals.

The UMLS Knowledge Source Server (UMLSKS) is the set of machines, programs and Application Programmer Interfaces (APIs), written in Java, located and maintained by staff at the NLM that allow access to the UMLSKS services. This document describes the system architecture for this version of the UMLS Knowledge Source Server and supporting systems and tools. For the remainder of the document, UMLSKS will be used as shorthand for the UMLS Knowledge Source Server.

1.1 Requirements Summary

1.1.1 Key Requirements and Design Goals

The driving requirements for the UMLSKS are as follows:

- Public access to UMLS Knowledge Sources via a web interface.
- Programmer access to the same functions and features of the UMLSKS available from the web interface.
- 24x7 availability, with minimal down time.

Additional design goals:

- Extensibility: the system is expected to have a long life and should improve with the technology over time.
- Scalability: the system is expected to handle an increasing number of user requests and larger UMLS vocabularies.
- Ease of administration: NLM and its contractors should be able to operate and maintain the system with a minimum of effort.
- System and software upgrades should generally not require any interruption of service, although moderate performance degradation is acceptable.

1.2 External Interfaces

The UMLSKS provides three mechanisms for external entities to interface with the UMLSKS. The first is through a web server running on a local machine at the NLM; the second is through an Application Programmer Interface (API) that connects user programs to the UMLSKS; and the third is a TCP/IP socket interface for non-Java programs to access the services of the UMLSKS. The web interface provides links to several web sites providing related information and services. The National Library of Medicine (NLM), National Institutes of Health (NIH), Health and Human Services (HHS) and other government organizations maintain these sites.

1.3 Performance, Sizing and Loading

The UMLSKS system is expected to occupy approximately 200 MB of disk space per UMLS release. The RAM required to efficiently run the UMLSKS will be determined during system testing but is expected to be on the order of 4 GB. Load balancing and changes to the machine configurations may be made after initial testing is complete.

1.4 Logical Architecture

The logical system architecture for the Knowledge Source Server is shown at a high level in Figure 1. The services of the UMLSKS can be accessed three different ways: through a web client using a standard browser (Netscape or Internet Explorer), through a program written to use the UMLSKS API, or through a TCP/IP socket-based interface. Data is returned to the user in XML form. A set of classes that provide a data-centric representation of the UMLS Metathesaurus, Semantic Network, and SPECIALIST Lexicon is provided with the API and is capable of reading the XML generated by the API methods (except the **`query`** method).

The web interface issues HTTP requests through the Internet to a web server. The web server then issues a request using Java's Remote Method Invocation (RMI) methodology to execute a particular function on behalf of the user. The RMI Server receives the request, executes the request against the database, and formulates and returns that result to the web interface in XML.

The API issues RMI requests through the Internet directly to the RMI server using the RMI protocol. The RMI Server receives the request, executes the request against the database, and formulates and returns the result to the client API in one of two forms. The client program can request the information be returned in XML form and may subsequently use the Object Model classes to interpret the data as a set of data-centric objects. The object form is a set of classes that can be incorporated into the client application that provide functions for directly manipulating the database request results from within a Java program.

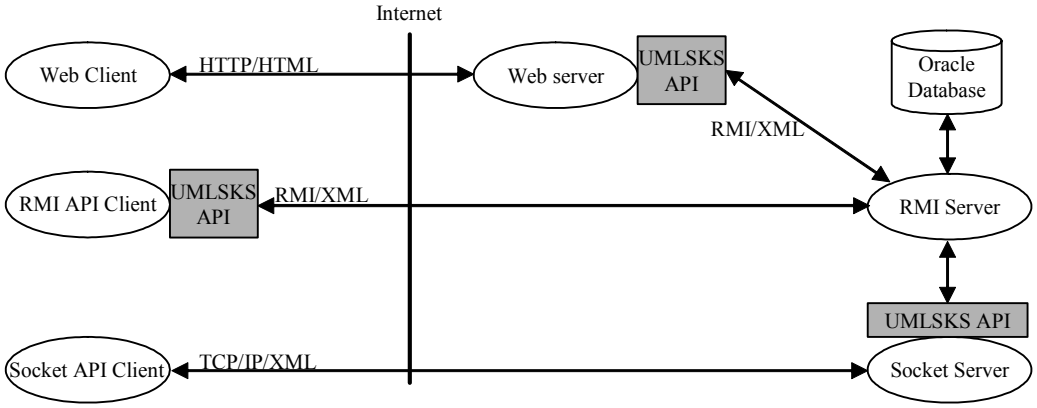


Figure 1 - Logical System Diagram

1.5 Physical Architecture

The four machines shown in Figure 2 will be used for the UMLSKS.

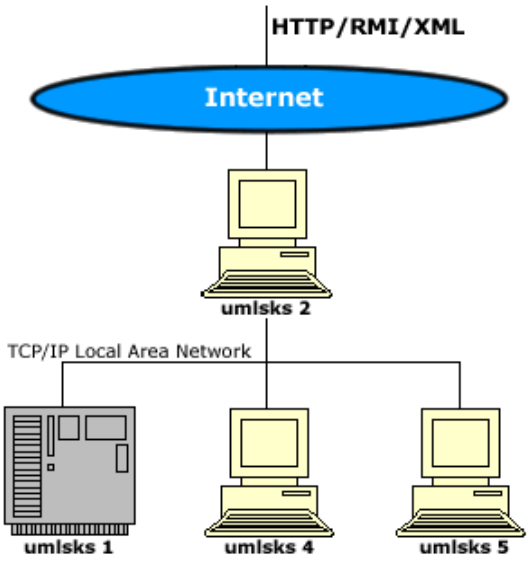


Figure 2 - Physical System Environment

The table shown below shows the allocation of processing function to UMLSKS machines.

Machine	Hardware Configuration	Functions
umlsks1	Enterprise 4000, 4 CPU Sun Microsystems machine, 2 GB RAM	- Resonate Back-up Server - Web Server - UMLSKS Server/Oracle Database
umlsks2	Sun Sparc Ultra 2 workstation	- Primary Resonate Server - MySQL User Accounts Database
umlsks4	Sun Sparc Ultra 2 workstation with 2 GB RAM	- Web Server - UMLSKS Server/Oracle Database
umlsks5	Sun Sparc Ultra 2 workstation with 2 GB RAM	- Web Server - UMLSKS Server/Oracle Database

Table 1 – Physical Machine/Processing Function Map

2 Using the UMLSKS Web Interface

The UMLSKS Server is typically accessed using a standard web browser such as Netscape or Internet Explorer. This chapter describes usage of the UMLSKS web interface. Because of differences in the way browsers present information, the displays actually seen may be slightly different from those shown in this guide.

2.1 Obtaining Access to the UMLSKS

The UMLSKS Server and UMLS data sources are restricted resources that require compliance with the copyright restrictions of the NLM and the UMLS. Users request access accounts from the main page.

NOTE: Prior to creating a UMLSKS user account, users must have obtained and signed a valid license agreement with the NLM. TO obtain a license for access to UMLS data, click on the link http://www.nlm.nih.gov/research/umls/license.html and follow the instructions found on that page to obtain a valid license number for use during the account creation process.

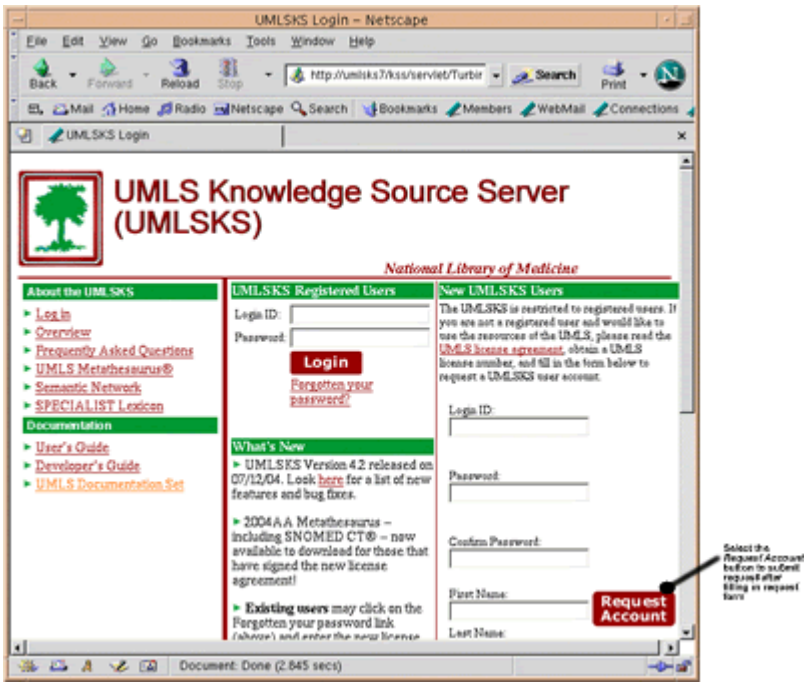


Figure 3 – UMLSKS Request Account Area

Fill in the form on the right side of the display and press the Request Account button. Your input will be validated, and if properly entered, your account will be created. You may then access the UMLSKS services by logging onto the UMLSKS.

2.2 Logging into the UMLSKS

Login to the UMLSKS requires users to have a valid account name and password (See Section 2.1 for instructions on obtaining and account). To log in, fill in your name and password in the login area and press the *Login* button.

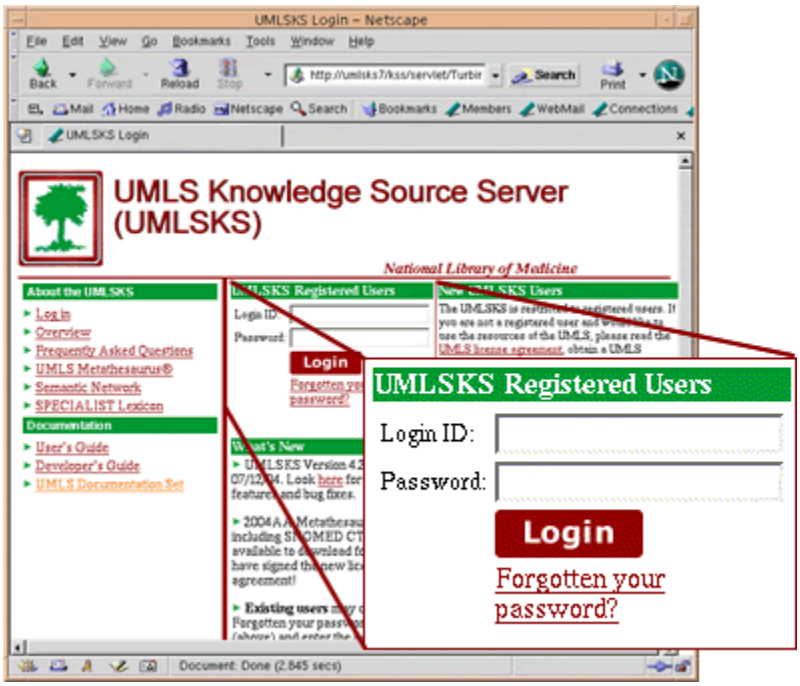


Figure 4 – UMLSKS Login Area

If you have forgotten your password, you can request the UMLSKS to update the password to a new one by pressing the *Forgotten Your Password?* link found under the login area shown above. You will be requested to enter information about your account name, your UMLS License number and your UMLS Licensee's name. The new password you enter on this screen will take effect immediately and you may login to the UMLSKS using the new password.

2.3 Navigating the UMLSKS Interface

Once you are properly logged in you will see the UMLSKS home page shown in Figure X. This page is continuously updated as changes occur to the UMLSKS software. In general, the left-side menu remains fixed for access to downloads, documentation, and other UMLSKS resources. The middle section of the home page is dedicated to quick searching of the three UMLS data sources and to updating users on newsworthy items. The right side of the screen allows access to the advanced searching capabilities associated with each of the UMLS data sources.

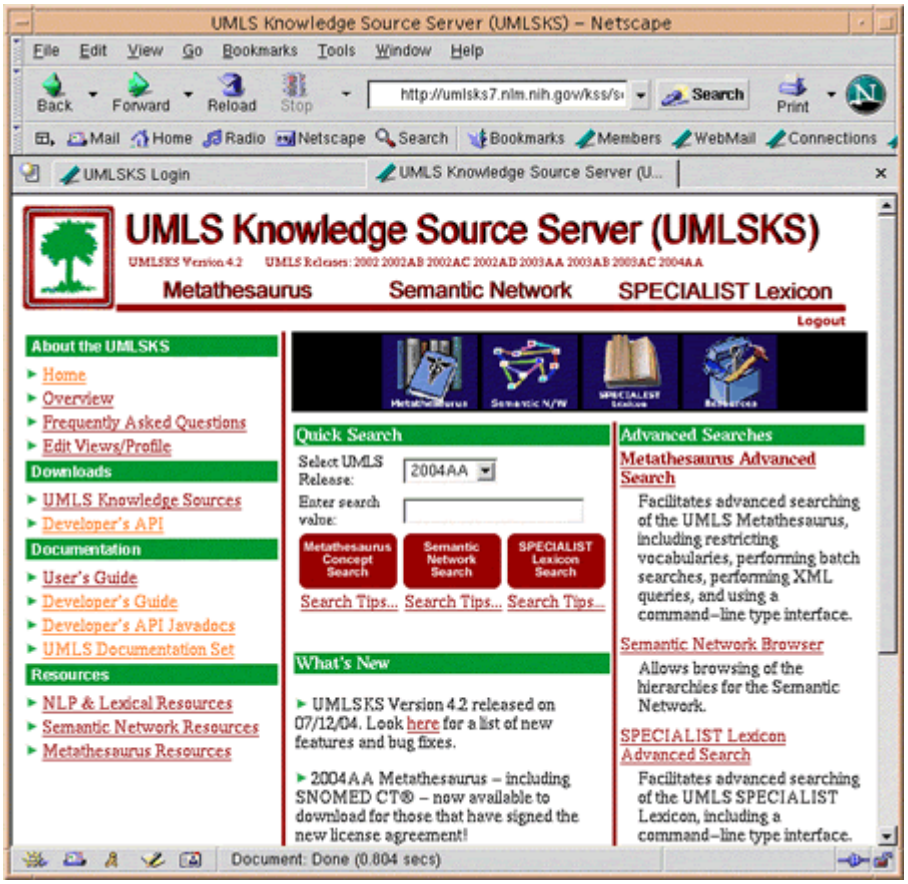


Figure 5 – UMLSKS Home Page

The **What's New** section of the home page, found in the lower middle section of the page, describes the news items for the UMLSKS. These items may reflect scheduled downtimes for the system, expected updates to the software, and any other newsworthy information.

2.3.1 Downloading UMLS Knowledge Sources

The UMLSKS provides the capability to download the UMLS Knowledge Sources in various machine formats. To access the download page, click on the *UMLS Knowledge Sources* link under the **Downloads** topic.

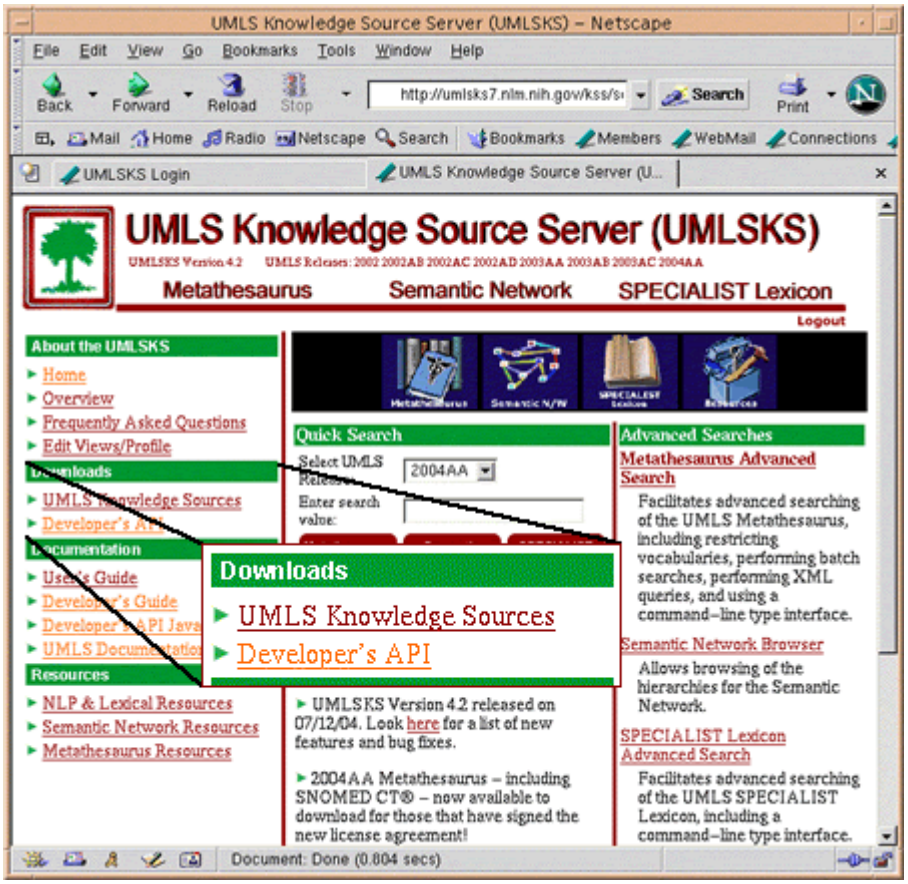


Figure 6 – Download Topic Menu

Once selected, the page identifying the UMLS releases that are available for download will be displayed. Follow the instructions given in each page to download the desired UMLS release.

2.3.2 Downloading UMLSKS Application Programmer Interface (API)

The UMLSKS allows application developers to develop programs that take advantage of the UMLSKS search capabilities. The download of the API, associated data and sample files can be accessed by clicking on the *Developer's API* link under the **Downloads** topic. The displayed page describes the steps required to download the current version of the UMLSKS API and how to setup your environment for proper installation and configuration of the API samples. Follow the instructions given on the page for detailed steps in completing the API download.

2.3.3 Viewing UMLSKS Documentation

The UMLSKS comes delivered with a complete set of documentation including references for using the system and other UMLS resources. The left-side menu contains a list of the available documentation for the UMLSKS. Each entry in the list is a link that displays the document's contents. For example, this User's Guide may be viewed in sections on-line by clicking on the *User's Guide* link.

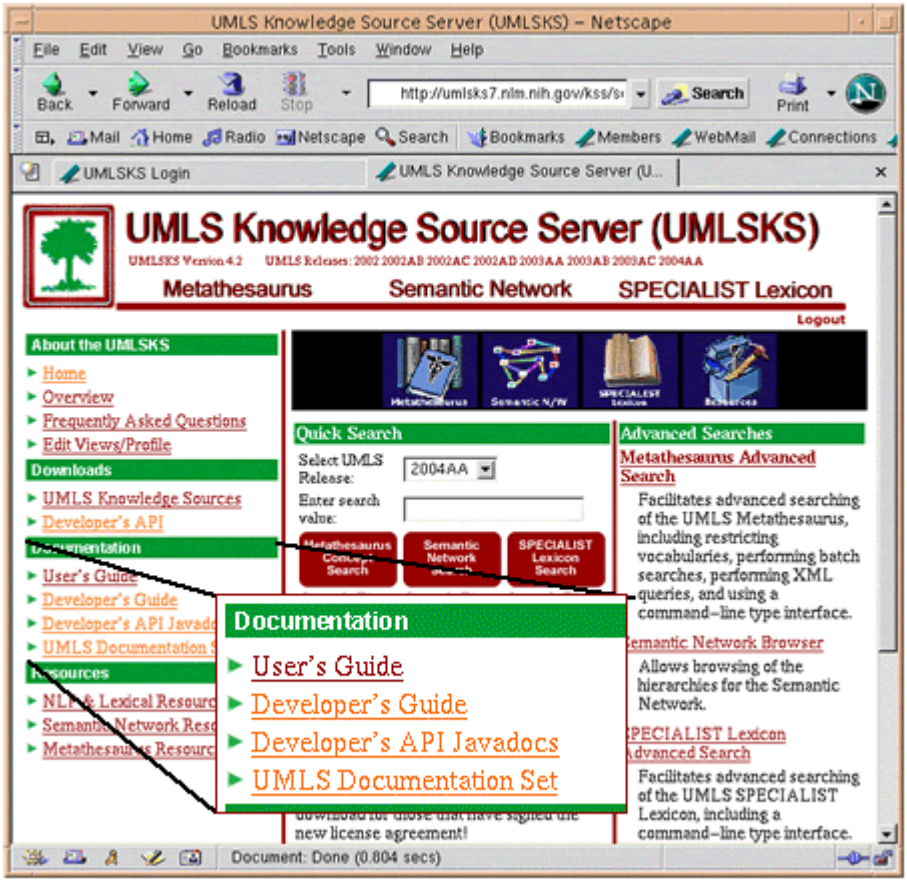


Figure 7 – Accessing UMLSKS Documentation

2.3.4 Accessing UMLSKS Resources

The UMLSKS provides access to additional resources developed at the National Library of Medicine that may be useful for UMLS research and development. These resources are categorized and listed under the Resources topic. Each entry under the topic is a link that displays information on the selected topic. For example, tools and resources associated with NLP and lexical processing can be displayed by clicking on the link labeled *NLP & Lexical Resources*.

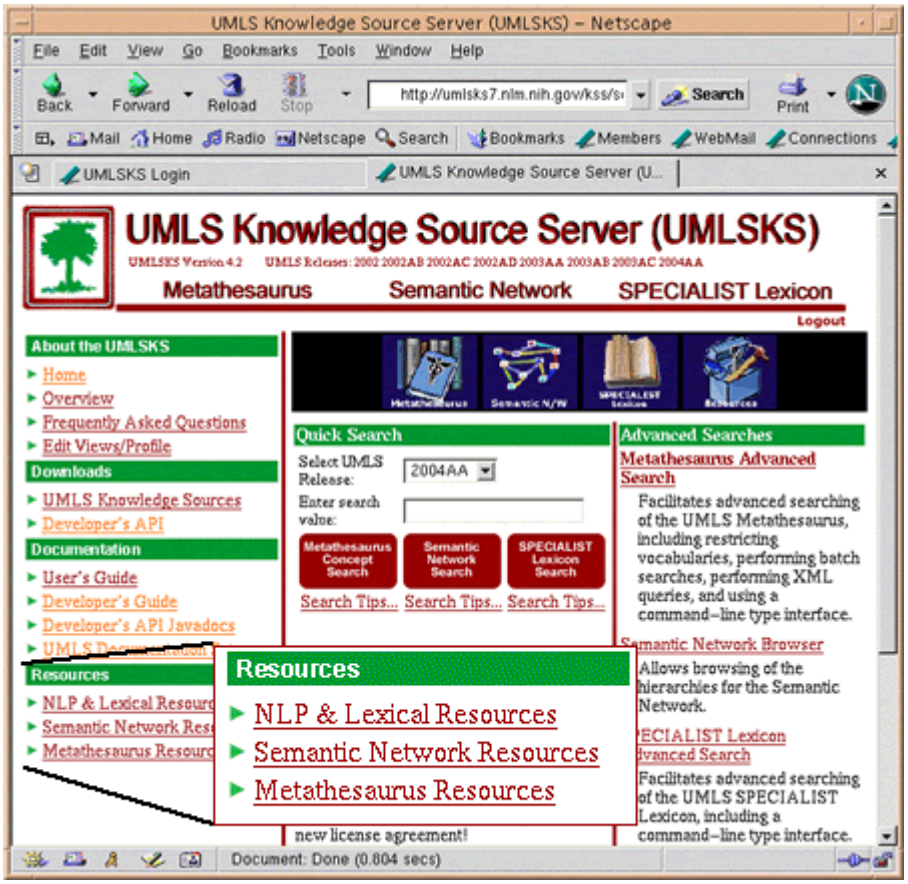


Figure 8 – Accessing UMLSKS Resources

2.3.5 Performing Quick Searches

The middle section of the home page provides quick access to the basic search functions of each of the UMLS Knowledge Sources. By entering a search string in the text entry box and clicking on the appropriate button (*Metathesaurus Search*, *Semantic Network Search*, or *SPECIALIST Lexicon Search*), the corresponding knowledge source will be searched and the search results will be displayed.

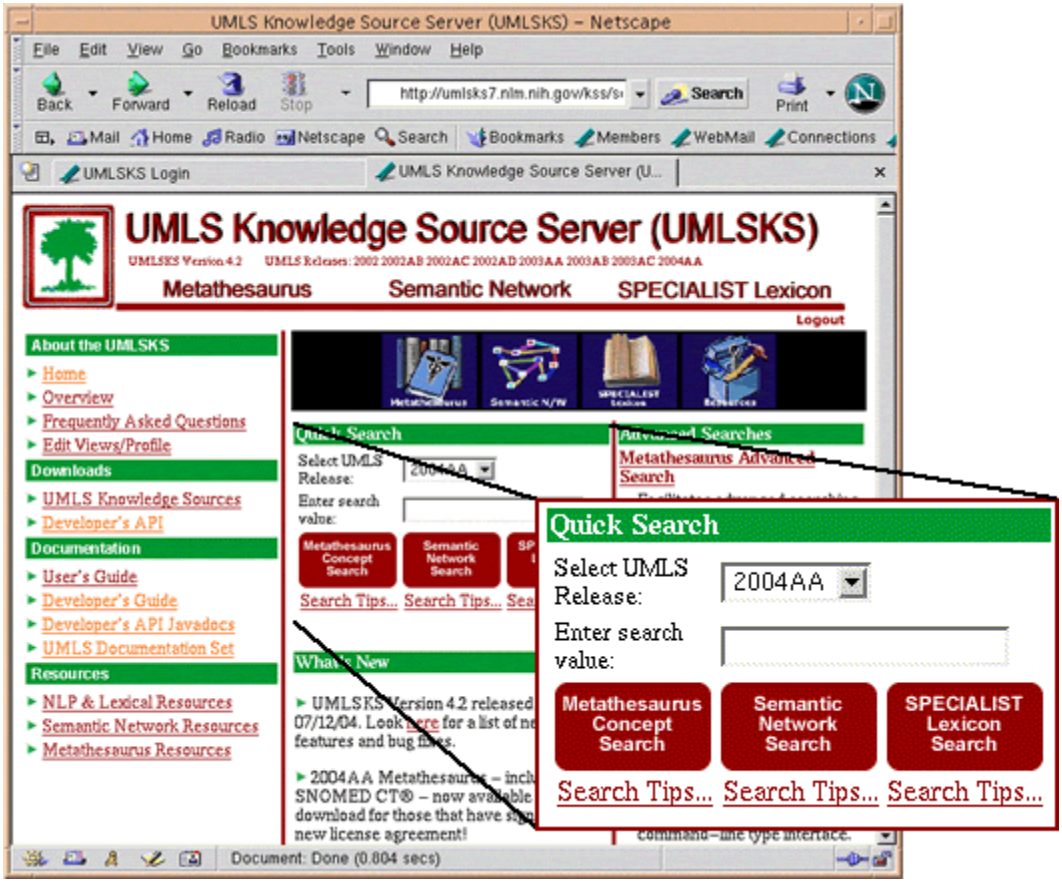


Figure 9 – Performing Quick Searches

2.3.6 Advanced Searching of UMLS Knowledge Sources

The right side section of the home page provides quick access to the advanced search functions of each of the UMLS Knowledge Sources. Clicking on these links displays the advanced search windows for each of the knowledge sources.

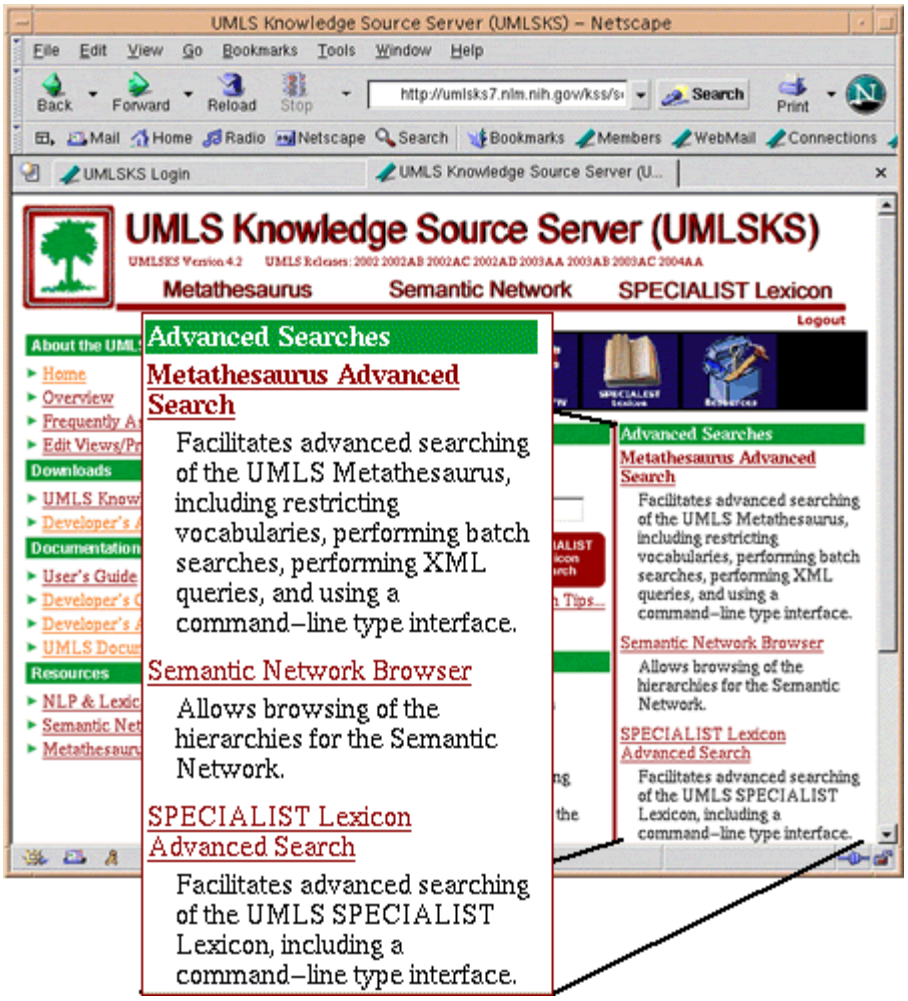


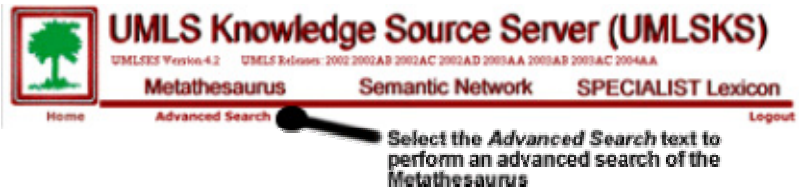
Figure 10 – Advanced Searching of UMLS Knowledge Sources

2.4 Accessing the Metathesaurus

The Metathesaurus is a UMLS resource that provides detailed information on concepts and terms across multiple vocabularies. This resource may be searched using the web browser and information related to the user's request would be displayed. There are four ways to search the Metathesaurus:

- Enter terms/CUIs into the basic, quick searching from the home page after logging in as described in Section 2.3.5.
- Select *Metathesaurus Advanced Search* from the home page after logging in as described in Section 2.3.6.
- Select the large tab marked *Metathesaurus* for basic searching when it appears along the top navigation area.

- Select the small tab marked *Advanced Search* for advanced searching when it appears along the top navigation area.



2.4.1 Basic Searching

The Metathesaurus can be searched for concepts and related information using the Basic Search capability. By selecting the Metathesaurus tab, the user will view a display allowing the user to enter a request to view concepts.

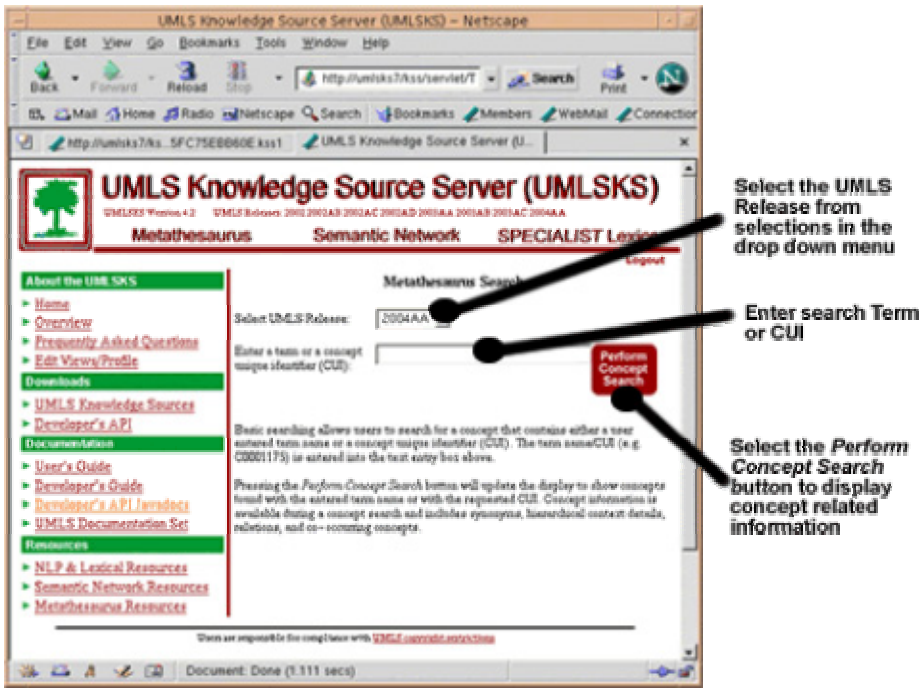


Figure 11 – Basic Searching of the Metathesaurus

Select the UMLS Release to search using the drop down menu of selections. Enter either a term name or a CUI in the text field. The system will automatically determine whether a CUI or term was entered based upon your input. Once this information is entered, select the *Perform Concept Search* to obtain the concept related information.

2.4.2 Advanced Searching

The Metathesaurus can also be searched using some advanced options, such as:

- Limiting the UMLS source for the data and specifying a search criteria upon the entered term name (normalization, truncation, etc.),
- Requesting execution of an XML query.

There are two ways to access the Metathesaurus advanced searching features:

- Select *Metathesaurus Advance Search* from the home page after logging in as described in Section 2.3.6.
- Select the small tab marked *Advanced Search* for advanced searching when it appears along the top navigation area.

The *Advanced Search* window appears and lists the advanced searching options as shown in the figure below.

After selecting the Advanced Search capability, the advanced searching window will appear.

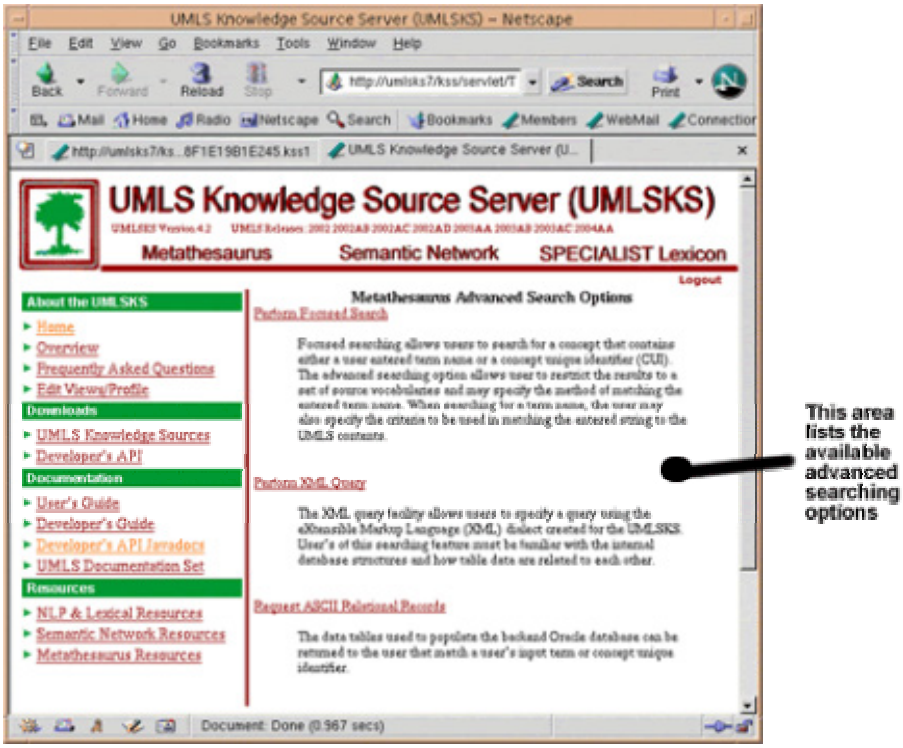


Figure 12 – Options for Metathesaurus Advanced Searching

The advanced searching window will display the list of available searching options with a brief description of each option. To perform a focused search, select the *Perform Focused Search* link. The *Perform XML Query* link displays the searching option of submitting an XML query to the UMLSKS for execution. The *Request ASCII Relational Records* option allows users to display a list of the original table records.

2.4.2.1 Focused Searching

The focused search of the Metathesaurus allows users to limit the data to be returned to those within selected sources and limited by the searching criteria to be applied to entered term names or CUIs. The return data can be selected as either the concept related information or the term related details.

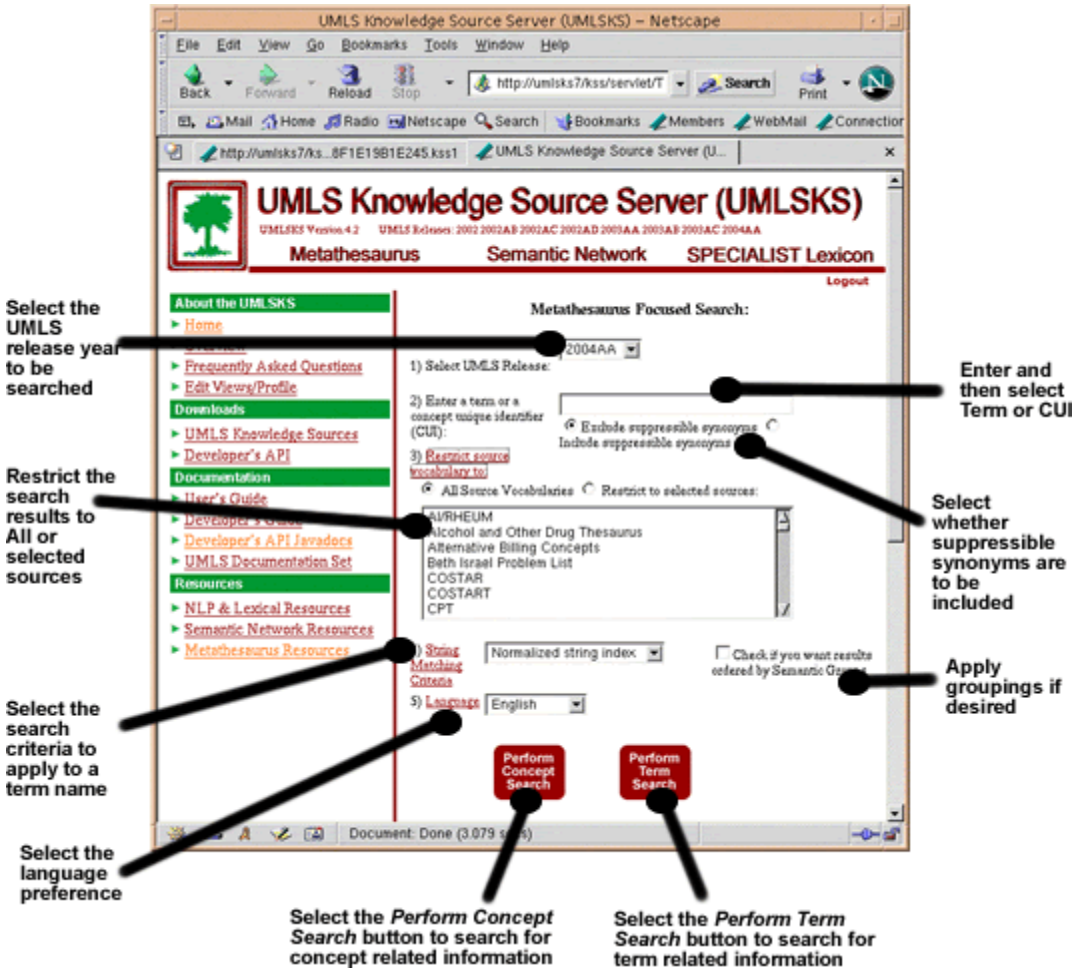


Figure 13 – Metathesaurus Focused Searching

To submit a focused search to the UMLSKS, select the UMLS release to search and enter the term name. Restrict the UMLS source to either all of the sources, or one or more sources from the list displayed. To restrict to all sources, select the radio button that reads *All Source Vocabularies*. To restrict to one or more of the sources in the list, select the radio button *Restrict to selected sources:* and select one or more sources from the list below the radio button. To "remove" a source from the list of sources, simply deselect that source by clicking on the source name.

If you entered a term, you may specify an option of how that term should be treated for searching purposes. Each option has a different effect on the way the search is performed and each search criteria option is described below. If you entered a CUI search, the options for string matching criteria are not used.

You may also specify that the list of returned concepts is to be sorted by semantic group. To enable this option, enable the checkbox labeled *"Check if you want results ordered by Semantic Groups."*

Once the search input data has been entered, select the *Perform Concept Search* to search for concept related information or select the *Perform Term Search* button to search for term related information.

2.4.2.1.1 Exact Match Search Criteria

The *Exact Match* search criteria will request that the input term or terms be matched exactly to the contents of the database.

- During a Concept search, only those concepts containing terms or synonyms that exactly match the input terms excluding any capitalization but including any punctuation, will be displayed to the user.
- During a Term search, only those terms containing variants matching the input terms exactly, excluding any capitalization but including any punctuation, will be displayed to the user.

2.4.2.1.2 Normalized String Search Criteria

The *Normalized String* search criteria will request that the input term or terms be string normalized before matching those normalized strings to the contents of the database. The string normalization processing provided by the Lexical Variant Generation (LVG) utility are used to generate the normalized strings used as the database match strings.

- During a Concept search, only those concepts containing terms or synonyms that match the normalized string versions of the input terms will be displayed to the user, regardless of case in the input term(s).
- During a Term search, only those terms containing variants matching the input terms that match the normalized string versions of the input terms will be displayed to the user, regardless of case in the input term(s).

2.4.2.1.3 Normalized Word Search Criteria

The *Normalized Word* search criteria will request that the input term or terms be word normalized before matching those normalized words to the contents of the database. The word normalization processing provided by the Lexical Variant Generation (LVG) utility are used to generate the normalized words used as the database match strings.

- During a Concept search, only those concepts containing terms or synonyms that match the normalized word versions of the input terms will be displayed to the user, regardless of case in the input term(s).
- During a Term search, only those terms containing variants matching the input terms that match the normalized word versions of the input terms will be displayed to the user, regardless of case in the input term(s).

2.4.2.1.4 Word Search Criteria

The *Word* search criteria will request that the input term or terms be processed by the LVG utility's word handling feature before matching those "processed" words to the contents of the database.

- During a Concept search, only those concepts containing terms or synonyms that match the word versions of the input terms will be displayed to the user, regardless of case in the input term(s).
- During a Term search, only those terms containing variants matching the input terms that match the word versions of the input terms will be displayed to the user, regardless of case in the input term(s).

2.4.2.1.5 Left-Truncation Search Criteria

The *Left-Truncation* search criteria will request that the input term or terms be considered the right-most part of the names for the requested concepts or terms. For example, input "AID" would search

the database for all of those concepts or terms that have "AID" as the suffix for one of its terms or synonyms.

- During a Concept search, only those concepts containing terms or synonyms that end in the input string will be displayed to the user, regardless of case in the input term(s).
- During a Term search, only those terms containing variants matching the input terms that end in the input string will be displayed to the user, regardless of case in the input term(s).

2.4.2.1.6 Right-Truncation Search Criteria

The *Right-Truncation* search criteria will request that the input term or terms be considered the left-most part of the names for the requested concepts or terms. For example, input "AID" would search the database for all of those concepts or terms that have "AID" as the prefix for one of its terms or synonyms.

- During a Concept search, only those concepts containing terms or synonyms that begin with the input string will be displayed to the user, regardless of case in the input term(s).
- During a Term search, only those terms containing variants matching the input terms that begin with the input string will be displayed to the user, regardless of case in the input term(s).

2.4.2.1.7 Approximate Match Search Criteria

The *Approximate Match* search criteria will request that the input term or terms be processed by the Approximate Matching utility before matching those processed terms to the contents of the database.

- During a Concept search, only those concepts containing terms or synonyms that match the processed term versions of the input terms will be displayed to the user, regardless of case in the input term(s).
- During a Term search, only those terms containing variants matching the input terms that match the processed term versions of the input terms will be displayed to the user, regardless of case in the input term(s).

2.4.2.1.8 Source Vocabulary Restriction

The Metathesaurus may be searched for concepts and the results may be restricted to a set of source vocabularies. When restricting to a source, both the input side and the output side of the query are restricted. For example, if a concept has terms in two different vocabularies and you restrict to one of the vocabularies, then only the term found in that vocabulary will match and will be returned as the search results. This is true irrespective of all other options set during searching (e.g. suppressibility, string manipulations, etc.)

2.4.2.2 Performing an XML Query

The XML query capability provides access to all of the data within the UMLS database through definition of a search query in XML. The capabilities are far ranging and as such have been given a dedicated chapter in the User's Guide, Chapter 4.

2.4.3 Viewing Concept Details

A search performed on the Metathesaurus can result in one of three things happening.

- The query results indicating that the UMLS does not have information that matched your search criteria and has potentially offered spelling suggestions,
- The query resulted in a single concept or term that matched your search criteria, or
- The query resulted in multiple concepts/terms returned that matched the search criteria.

When multiple concepts are returned to the user, the following display is presented allowing the user to select the concept of interest. By selecting the concept, the details of that concept are updated and the options described in the following sections are available.

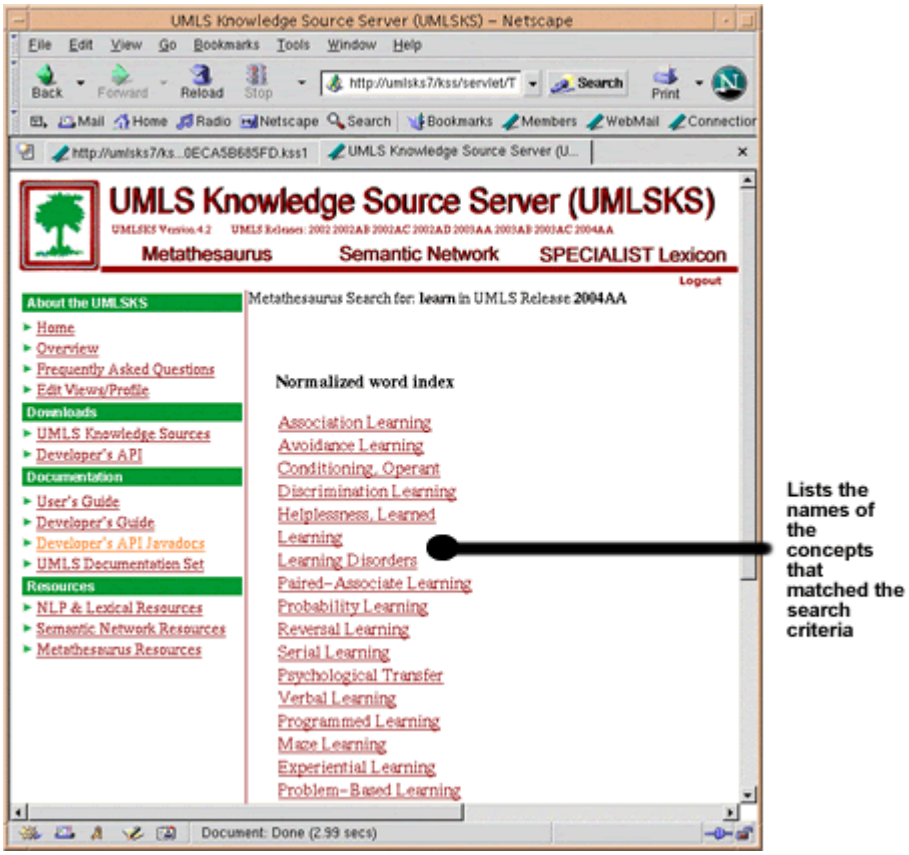


Figure 14 – Metathesaurus Concept List

2.4.3.1 Concept Details

The results of a concept query include details about the concept including:

- Semantic type,
- Synonyms for the concept (terms) including languages other than English,
- All definitions found for the concept,
- Any source vocabularies in which the concept/term appears,
- Hierarchical context,
- Associated expressions,
- Co-occurring concepts,

- Relations

Access to characteristics of the concept is handled by turning on/off the checkboxes on the left hand side of the concept window. As items are selected/deselected for display, the right hand side of the concept window is updated with the requested characteristic. In the diagram below, the search term "learning" was requested and the definitions, synonyms, and ancestors were selected using the corresponding left hand side checkboxes.

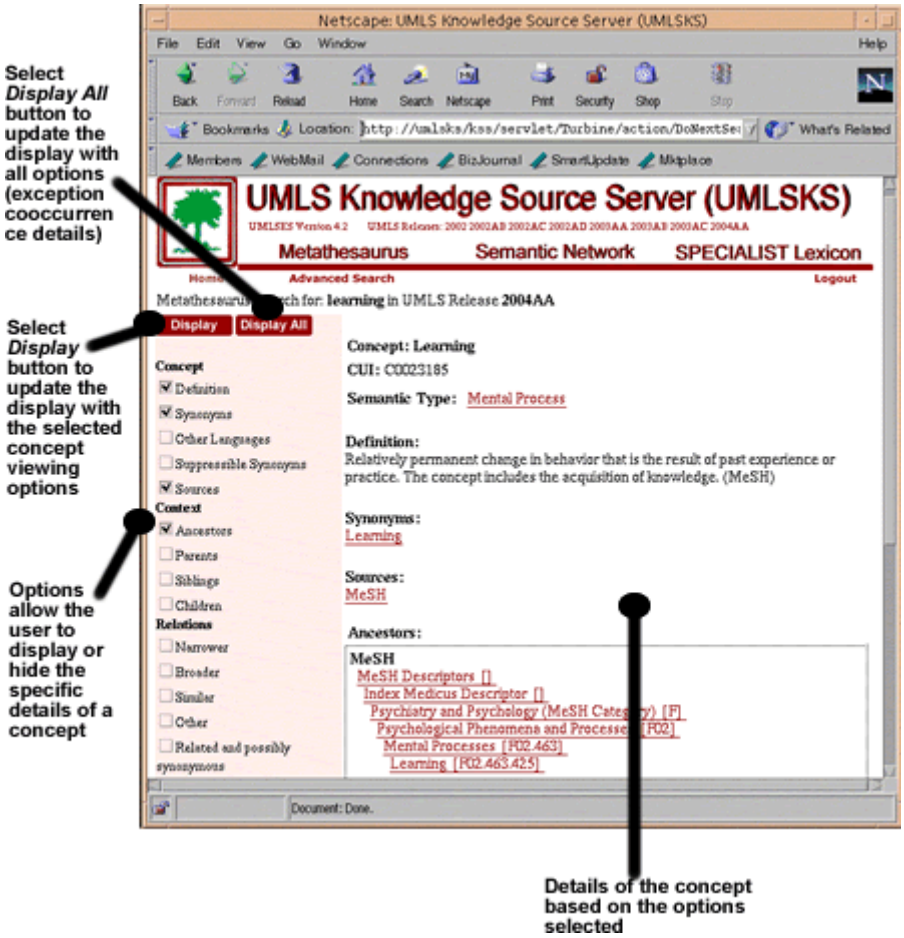


Figure 15 – Metathesaurus Concept Display

2.4.4 Viewing Term Details

A term search performed on the Metathesaurus can result in one of three things happening.

- The query results indicating that the UMLS does not have information that matched your search criteria,
- The query resulted in a single term that matched your search criteria, or
- The query resulted in multiple terms returned that matched the search criteria.

The results of a term query include details about the concept including:

- Variants for the term,
- MeSH entries, and

- String attributes.

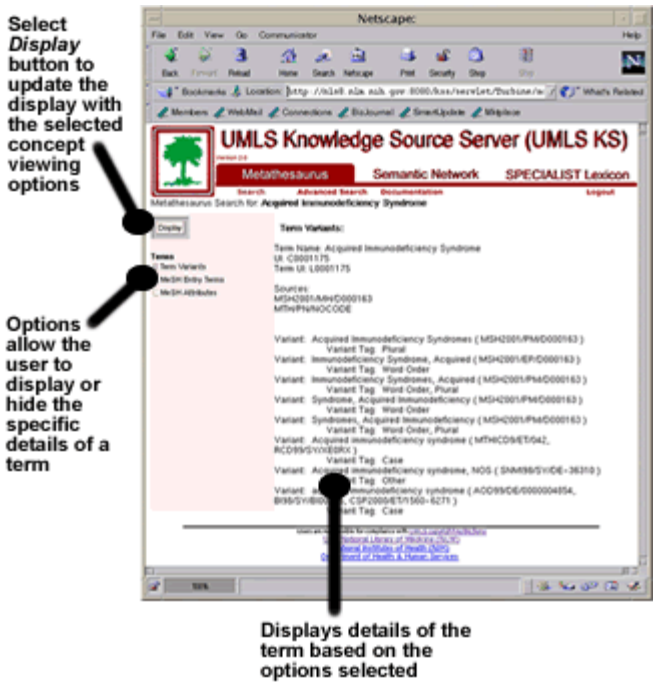


Figure 16 – Metathesaurus Term Display

Access to characteristics of the term is handled by turning on/off the checkboxes on the left hand side of the concept window. As items are selected/deselected for display, the right hand side of the concept window is updated with the requested characteristic.

2.4.5 Viewing Raw UMLS ASCII Records

The raw UMLS data used to populate the Oracle database can be displayed to the user. Select the link *Request ASCII Relational Records* from the Metathesaurus advanced search window. The window below is displayed. The drop down menu will show different tables based on the UMLS release year selected for display.

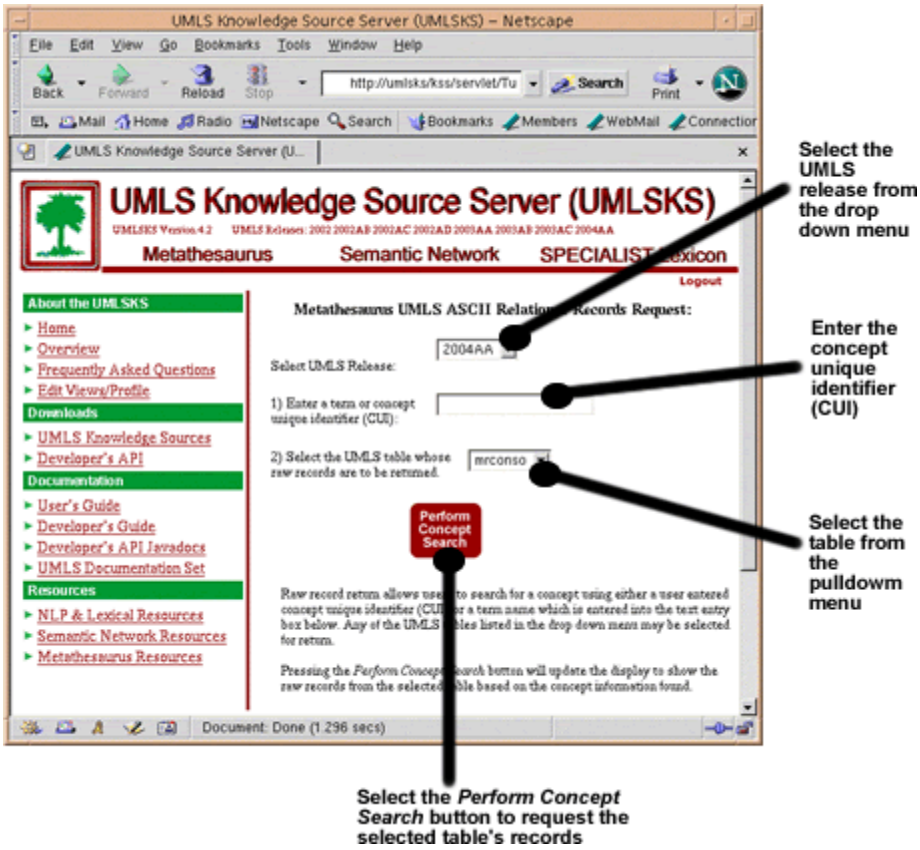


Figure 17 – Accessing Metathesaurus Raw ASCII Records

The output from the request of raw UMLS data is shown below for the search term **learning** and the table **`mrconso`** selected for UMLS release 2004AA.

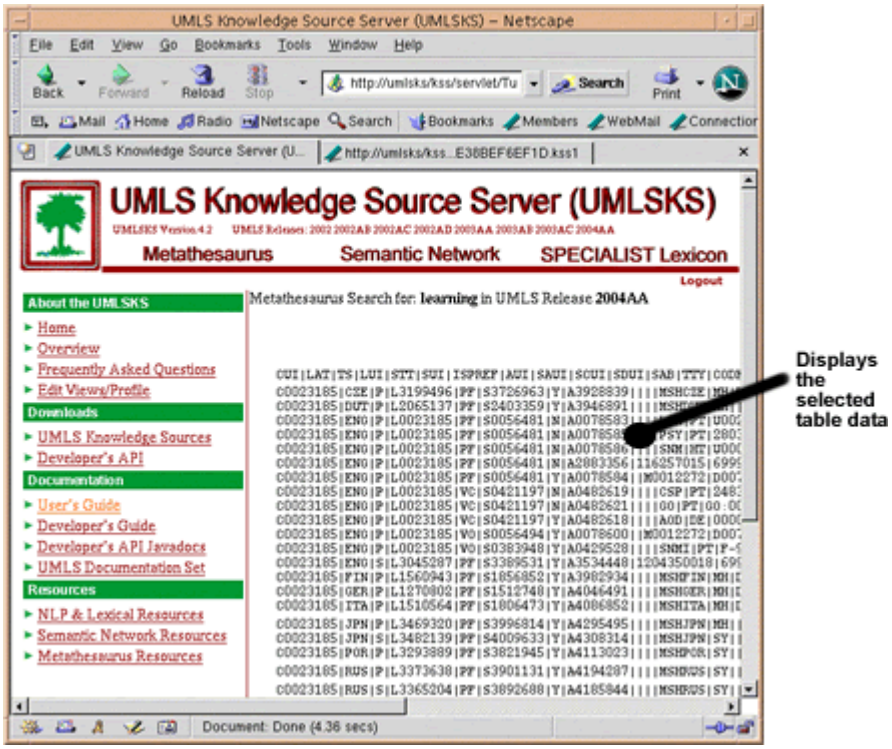
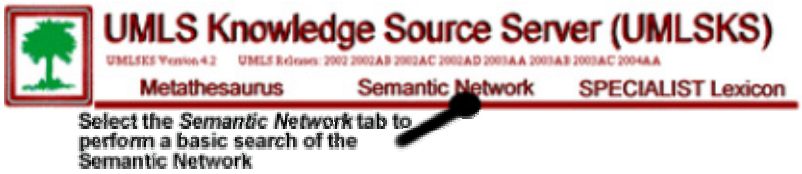


Figure 18 – Metathesaurus UMLS ASCII Records

2.5 Viewing the Semantic Network

The Semantic Network is a UMLS resource that provides detailed information about the semantic types found in the UMLS. There are three ways to search the Semantic Network.

- Enter terms into the basic, quick searching from the home page after logging in as described in Section 2.3.5.
- Select *Semantic Network Advanced Search* from the home page after logging in as described in Section 2.3.6.
- Select the large tab marked *Semantic Network* for basic searching when it appears along the top navigation area.



The Semantic Network may be browsed in two forms:

- HTML content
- Java Applet

The Java applet (Java Swing) version requires an applet to be downloaded to your local machine for the version to be viewed properly. This should happen automatically upon selecting to view the applet version. The HTML content version does not require any special components to operate properly.

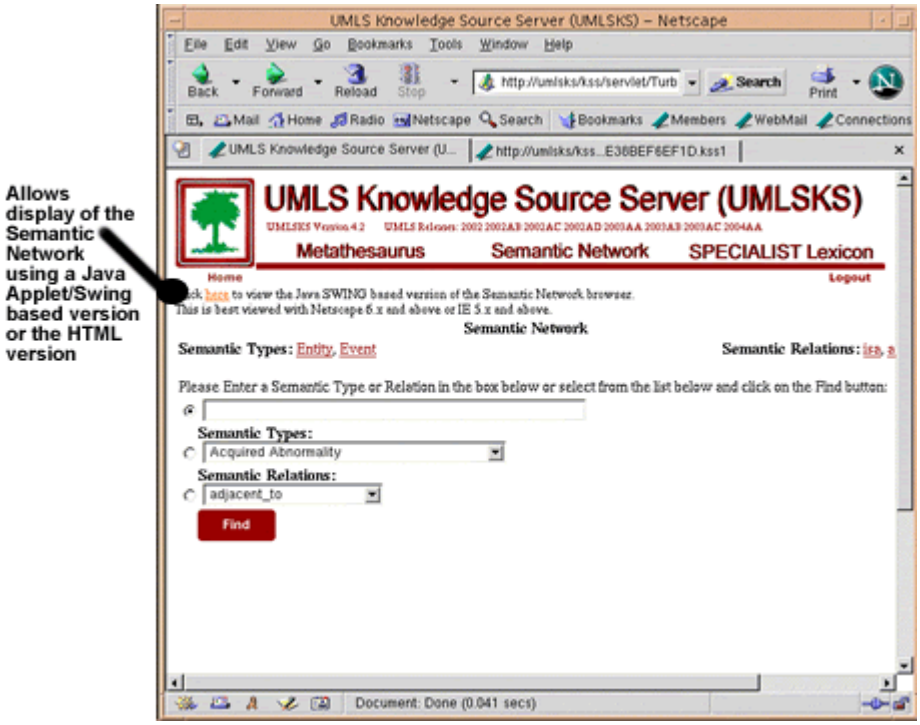


Figure 19 – Semantic Network Browser

The Semantic Network is divided into two categories: Semantic Types and Semantic Relations. The complete descriptions of these categories can be found at http://www.nlm.nih.gov/research/umls/META3.HTML. Once the Semantic Network is displayed users may search the Semantic Network for types and relations among data within the Metathesaurus.

2.5.1 Viewing a Semantic Type

Users may view a semantic type by either of two methods:

1) Entering a semantic type name in the text entry box
2) Selecting the semantic type root node **Event** or **Entity**.

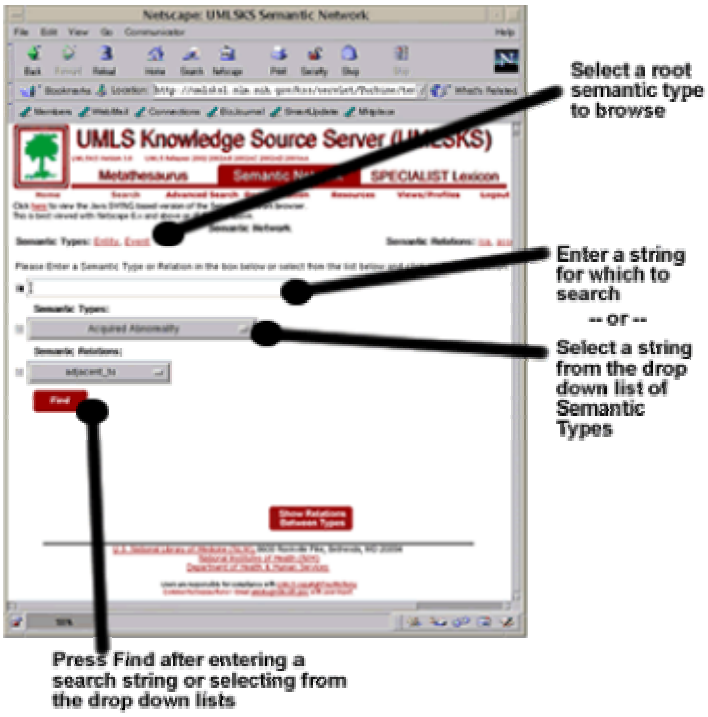


Figure 20 – Viewing a Semantic Type

Once the semantic type is entered or selected, the display updates to show the tree for the requested semantic type. In the example shown below, the **Event** root node was selected and the tree browsed for the semantic type **'Organism Function'**. The left side of the window shows the complete hierarchical tree structure for the root node selected. The selected semantic type is shown in bold red text. The right side of the display updates with the details about the selected semantic type.

Figure 21 – Browsing Semantic Types

2.5.1.1 Viewing Concepts with a Semantic Type

Metathesaurus concepts each are assigned one or more semantic types from the semantic network. The concepts with a particular semantic type can be viewed by selecting the **'Concepts'** checkbox once the semantic type has been selected from the left hand side tree.

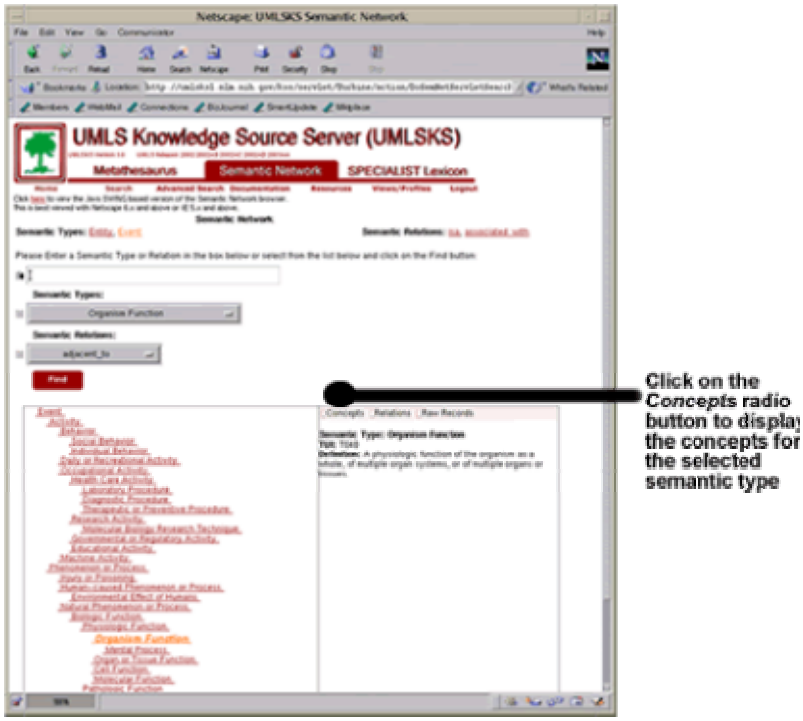


Figure 22 – Show Concepts with a Semantic Type

Once clicked, the right side of the display updates with the list of concepts belonging to the selected semantic type. Each entry is a link that will open another window with the Metathesaurus concept information displayed.

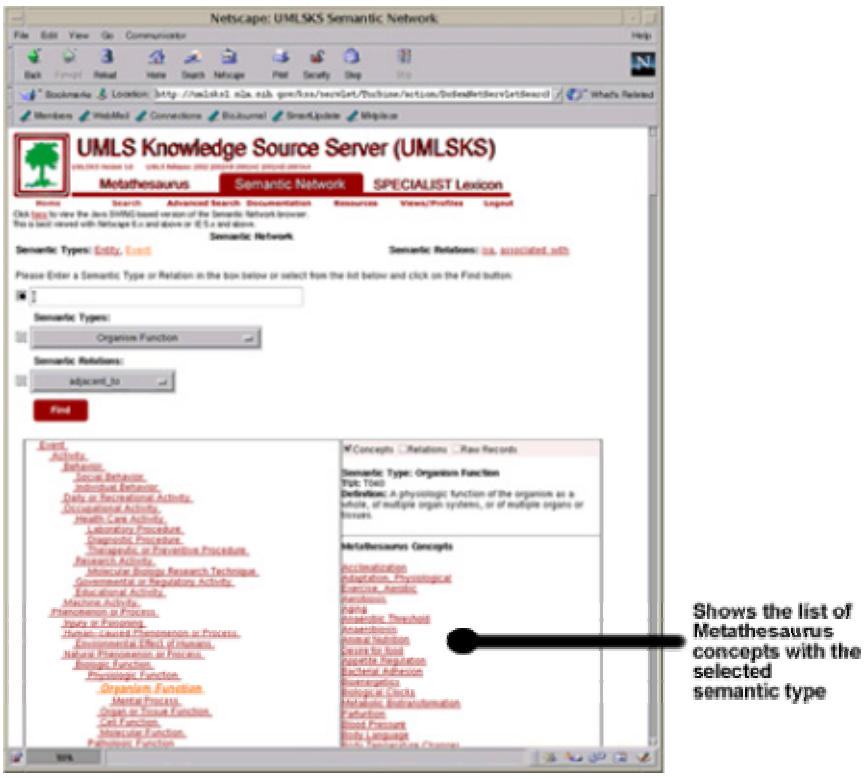


Figure 23 – Concepts with a Semantic Type

2.5.1.2 Viewing Semantic Relations in which a Semantic Type Participates

Semantic types within the network participate in one or more semantic relations with other semantic types. The relationships in which a specific semantic type participates can be viewed by selecting the **'Relations'** checkbox once the semantic type has been selected from the left hand side tree.

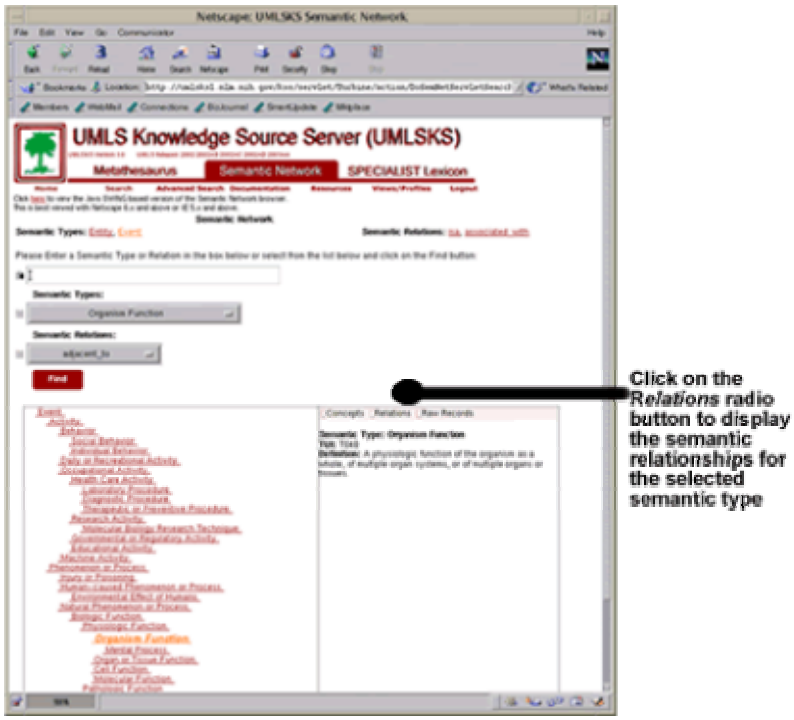


Figure 24 – Show Semantic Relations for a Semantic Type

Once clicked, the right side of the display updates with the list of semantic relations in which the selected semantic type participates. Each entry is a triplet of links describing the two semantic types that participate and the relationship that exists between them. Clicking on the appropriate link in the display area allows browsing of the semantic types and relations.

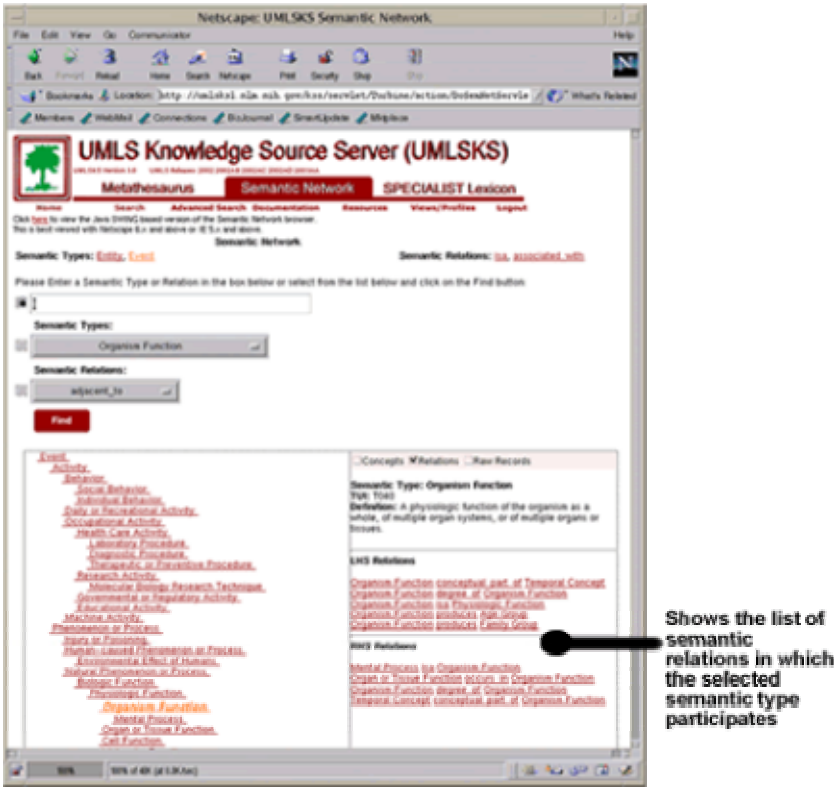


Figure 25 – Semantic Relations for a Semantic Type

2.5.1.3 Viewing ASCII Relational Records for a Semantic Type

The original ASCII relational records for a semantic type can be viewed by selecting the **'Raw Records'** checkbox once the semantic type has been selected from the left hand side tree.

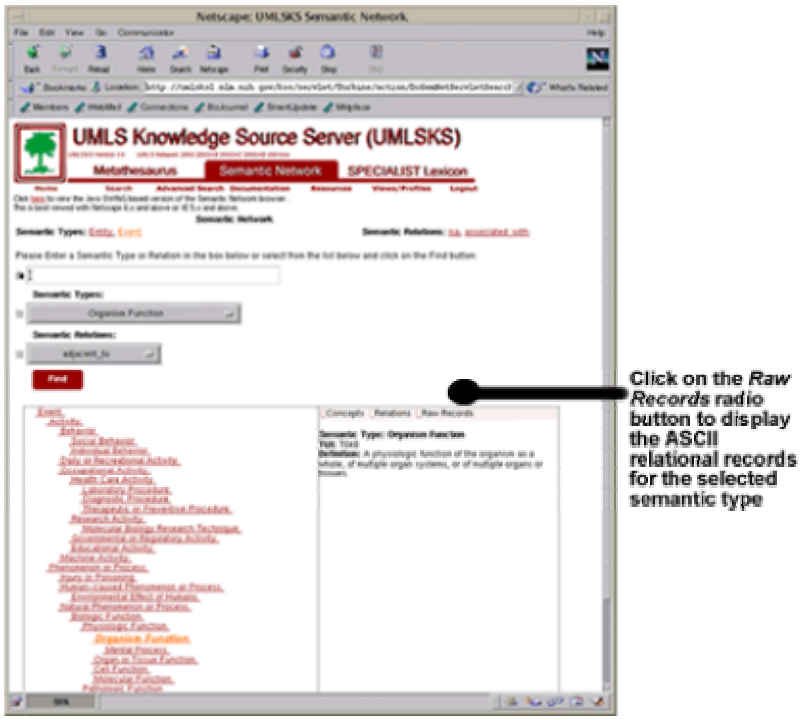


Figure 26 – Show ASCII Relational Records for a Semantic Type

Once clicked, the right side of the display updates with the ASCII records for the selected semantic type.

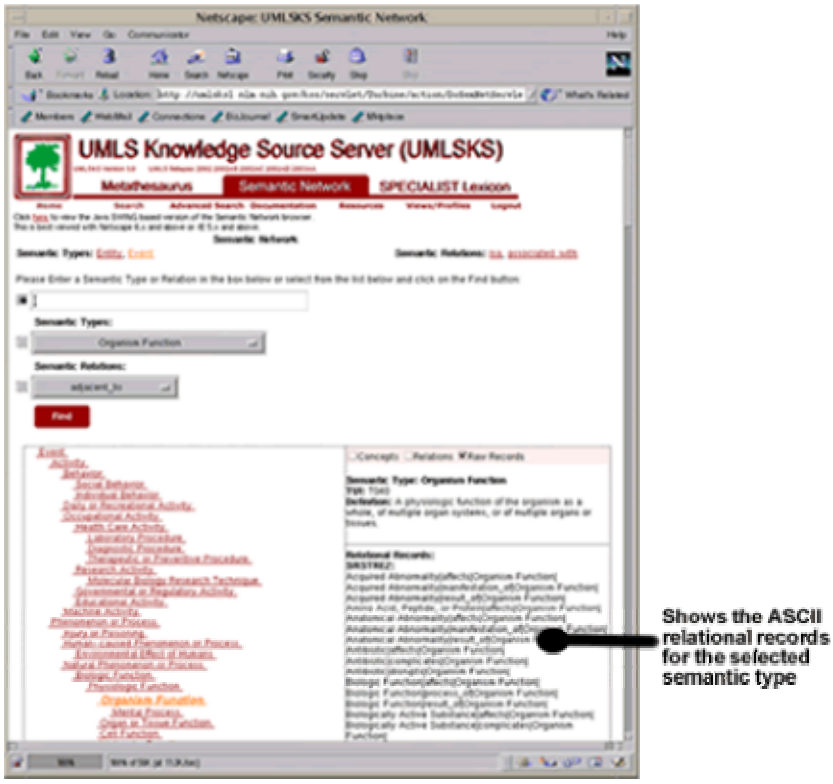


Figure 27 – ASCII Relational Records for a Semantic Type

2.5.2 Viewing a Semantic Relation

Users may view a semantic relation by either of two methods:

1) Entering a semantic relation name in the text entry box
2) Selecting the semantic relation root node **isa** or **associated_with**.

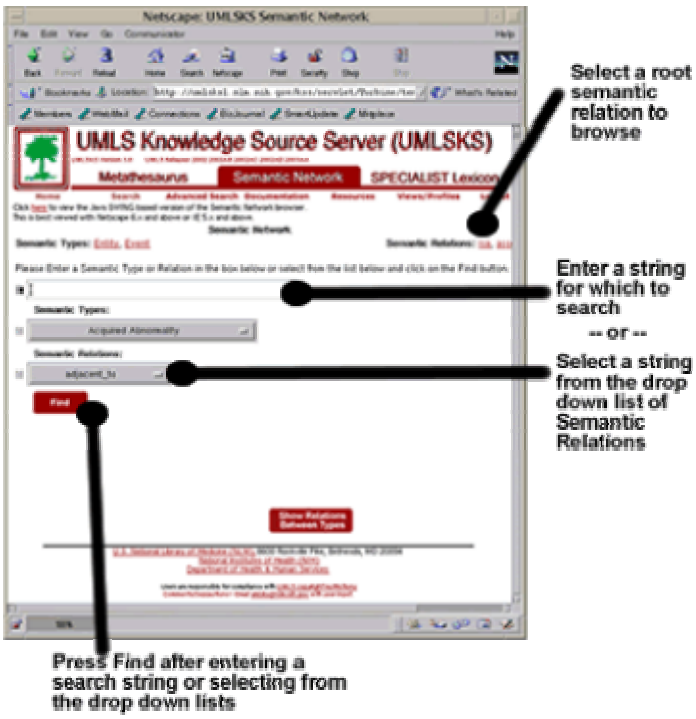


Figure 28 – Viewing a Semantic Relation

Once the semantic relation is entered or selected, the display updates to show the tree for the requested semantic relation. In the example shown below, the **associated_with** root node was selected and the tree browsed for the semantic relation **'precedes'**. The left side of the window shows the complete hierarchical tree structure for the root node selected. The selected semantic relation is highlighted in red. The right side of the display updates with the details about the selected semantic relation.

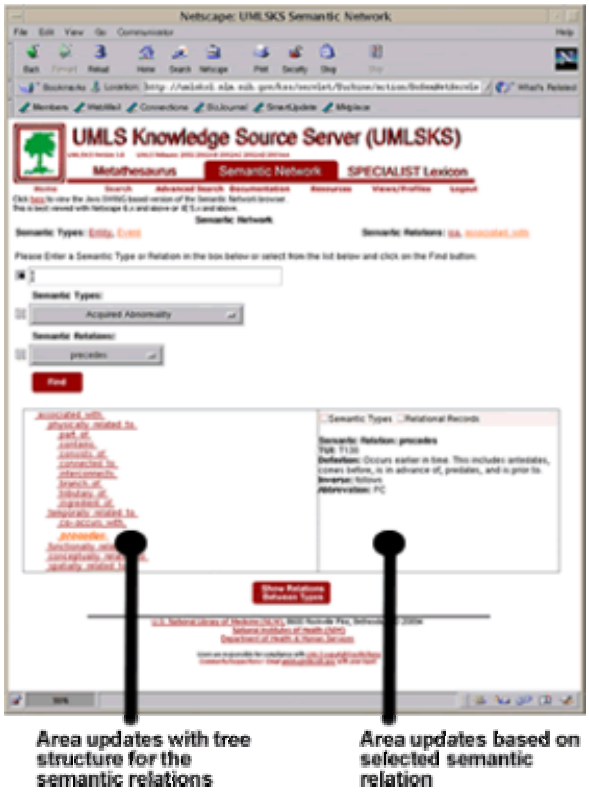


Figure 29 – Browsing Semantic Relations

2.5.2.1 Viewing Semantic Types that Participate in a Semantic Relation

Semantic types within the network participate in one or more semantic relations with other semantic types. The semantic types that participate in a given relation can be displayed by selecting the **'Semantic Types'** checkbox once the semantic relation has been selected from the left hand side tree.

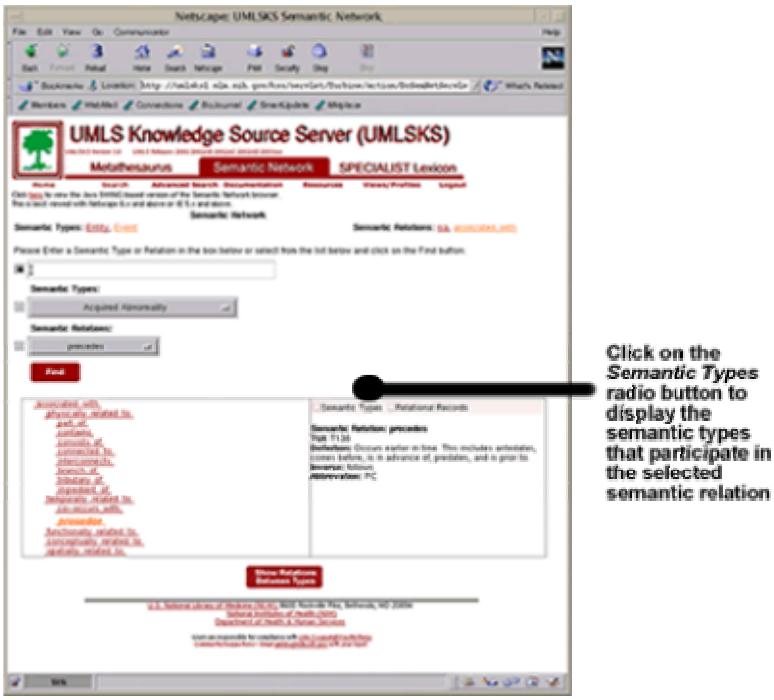


Figure 30 – Show Semantic Types that Participate in a Semantic Relation

Once clicked, the right side of the display updates with the list of semantic types that participate in the selected semantic relation. Each entry is a triplet of links describing the two semantic types that participate and the relationship that exists between them. Clicking on the appropriate link in the display area allows browsing of the semantic types and relations.

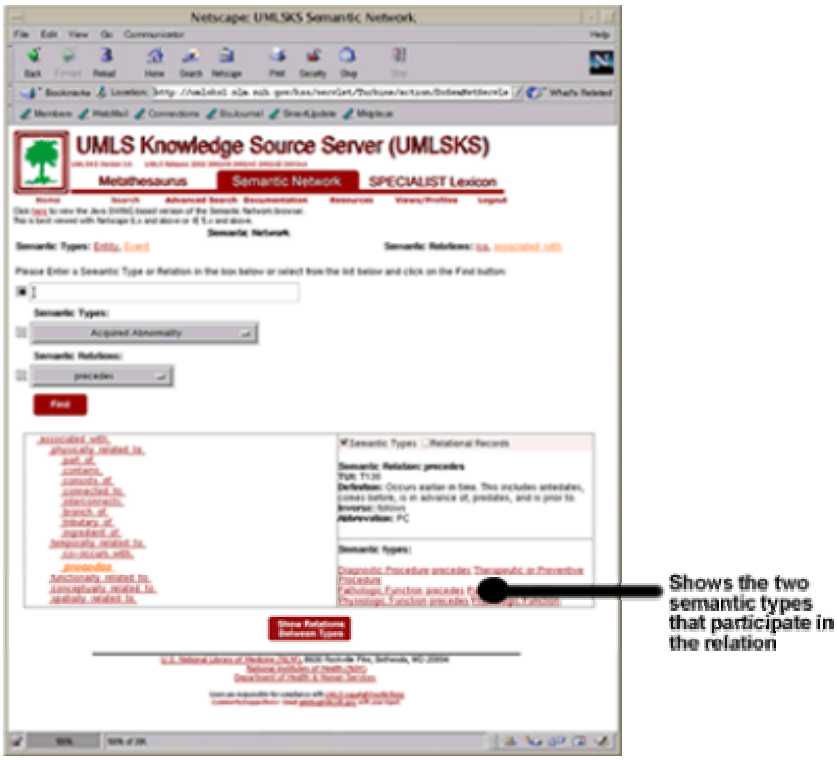


Figure 31 – Semantic Types for a Semantic Relation

2.5.2.2 Viewing ASCII Relational Records for a Semantic Relation

The original ASCII relational records for a semantic relation can be viewed by selecting the **'Relational Records'** checkbox once the semantic relation has been selected from the left hand side tree.

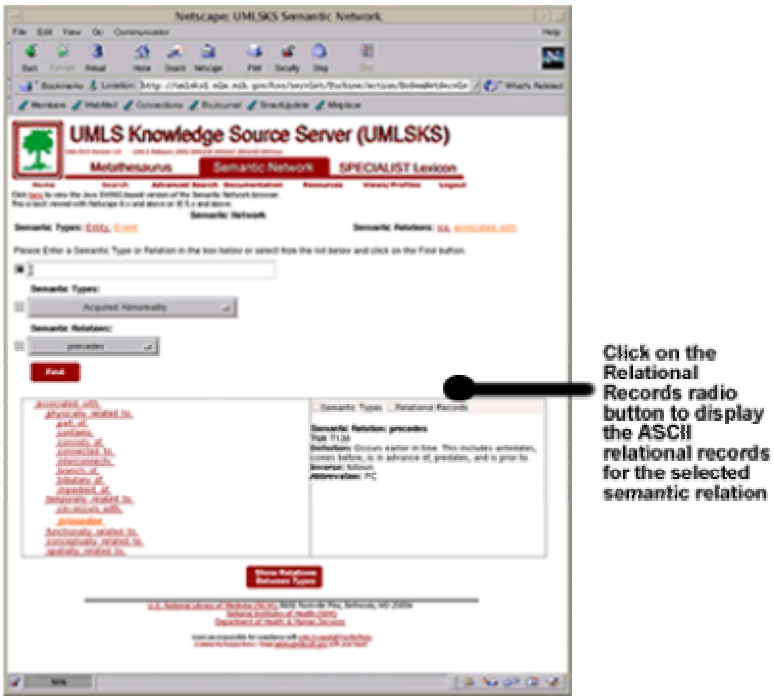


Figure 32 – Show ASCII Relational Records for a Semantic Relation

Once clicked, the right side of the display updates with the ASCII records for the selected semantic relation.

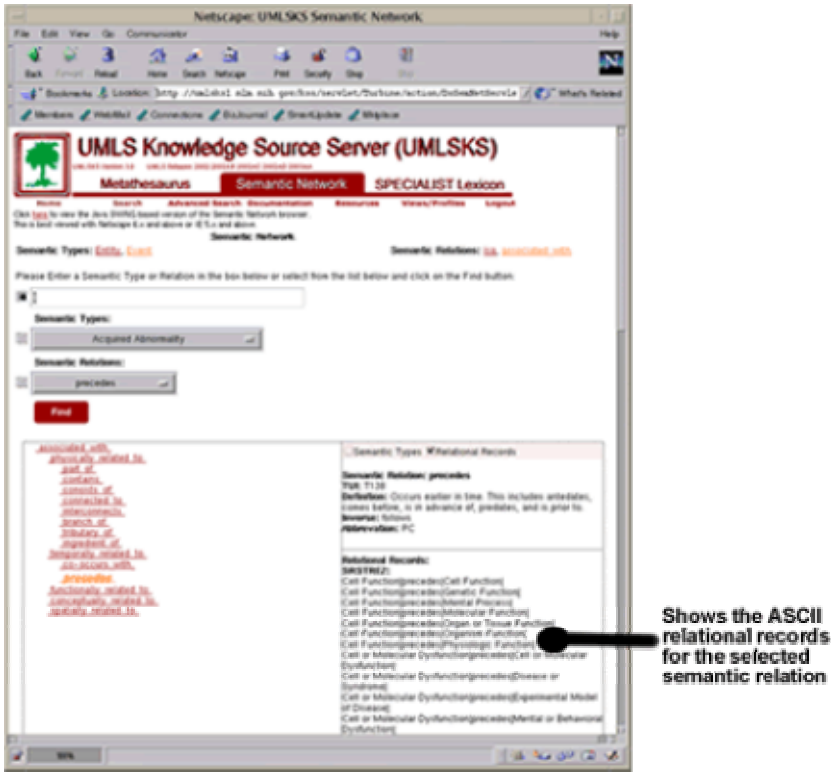


Figure 33 – ASCII Relational Records for a Semantic Type

2.6 Using the SPECIALIST Lexicon

The SPECIALIST Lexicon is a UMLS resource that provides lexical processing of input terms. There are three ways to search the Semantic Network.

- Enter terms into the basic, quick searching from the home page after logging in as described in Section 2.3.5.
- Select *Semantic Network Advanced Search* from the home page after logging in as described in Section 2.3.6.
- Select the large tab marked *SPECIALIST Lexicon* for basic searching when it appears along the top navigation area.

The window below will be displayed.

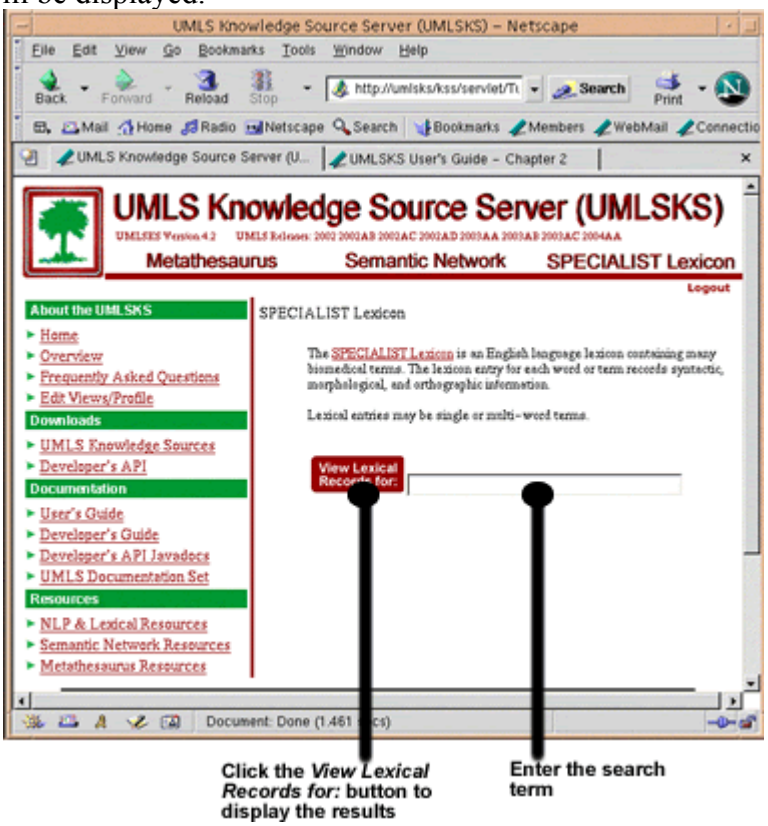


Figure 34 – Accessing the SPECIALIST Lexicon

Once displayed, users may perform lexical operations on user input terms. The input terms are placed in the text entry box and pressing the *View Lexical Records for:* button will update the display with the requested terms lexical variants. The picture below shows the result for the request for variants of **sleeps**.

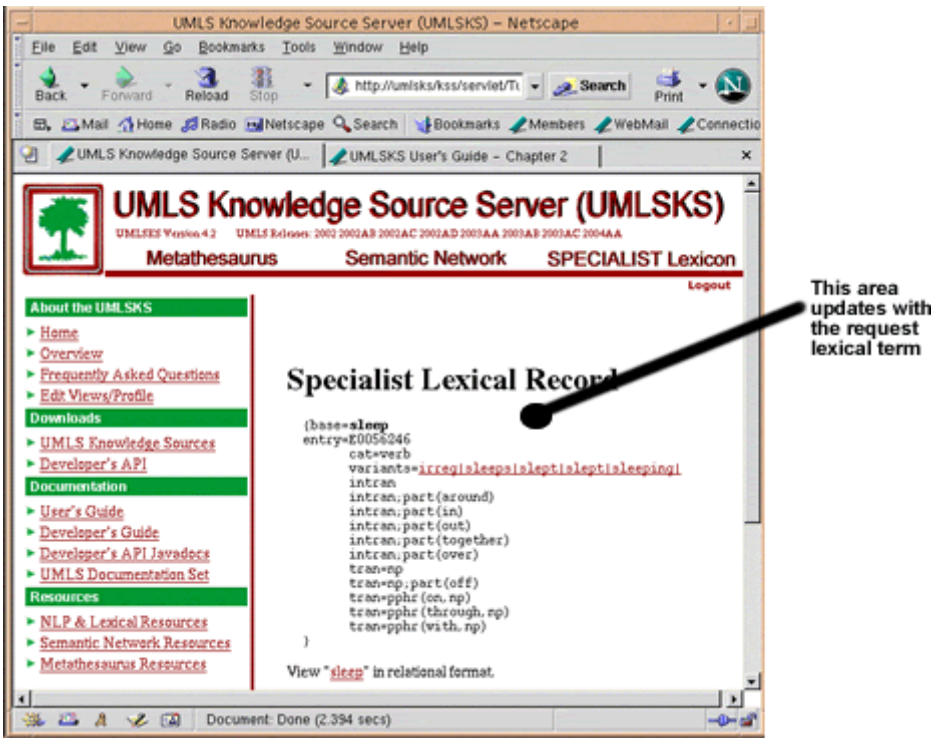


Figure 35 – Results from the SPECIALIST Lexicon

2.7 Updating Your View/Profile

The UMLSKS allows users to modify their user profile on file with the system. The user profile includes information that identifies you to the UMLSKS and may be updated at any time. To access your current profile, select the *Views/Profiles* button. You will be presented a form containing your current user profile information.

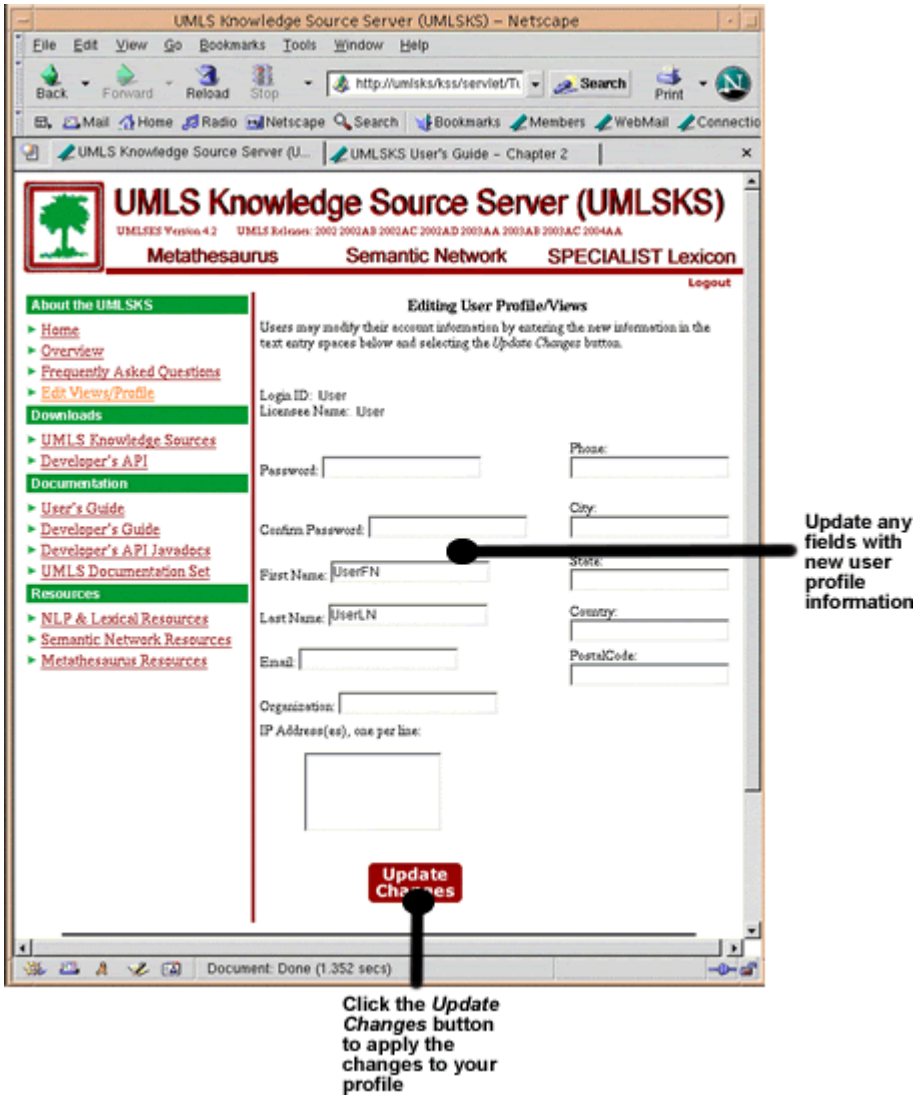


Figure 36 – Updating Your User Profile

Once displayed, you may modify any of the items shown. By clicking the *Update Changes* button, any changes you made will be saved by the system and your next login will use the modified profile.

2.8 Logging out of the UMLSKS

It is recommended that users log out of the UMLSKS once the user has completed his/her set of tasks. To log out, select the *Logout* button.

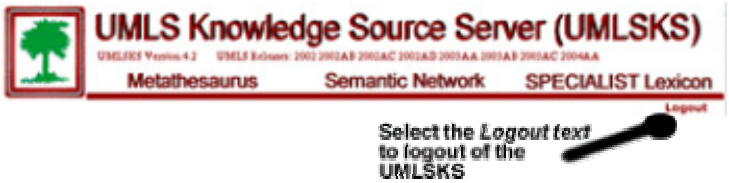


Figure 37 – Logging out of the UMLSKS

A window will be displayed indicating that you successfully logged out of the UMLSKS.

3 Using the UMLSKS Command-line Interface

The Command-Line Interface to the UMLSKS is also accessed using a standard web browser such as Netscape or Internet Explorer. Users must log on to the UMLSKS as described in Section 2.2 before the command-line interface options will become available. Because of differences in the way browsers present information, the displays actually seen may be slightly different than those shown in this guide.

The Command-line interface is slowly being phased out of normal operations. The interface accessing the Command-line functionality has been moved to the Metathesaurus resources section. To access it, click on the *Metathesaurus Resources* link found in the left-side menu and then locate the link *Perform Command-Line Query*. Batch processing of terms also uses the command-line functionality and can be found in the Metathesaurus Resources under the link *Perform Command-Line Query using File of Query Terms*.

3.1 Standard Options

There are a number of standard options that can be used in the command-line interface regardless of whether a concept or term search is performed. These include the ability to control the input of data for the search, the ability to control the output format for the search, or restricting the search to a list of UMLS sources.

All commands issued to the command-line interface must begin with the following:

```
ks -meta -c <TermOrCUI>
```

where `<TermOrCUI>` is the term or CUI to be requested. See sections 3.2 and 3.3 for details on the specific options for a concept search or a term search.

3.1.1 Controlling Command-Line Input

The command line interface allows users to specify an input type for the entry data. To request concept information based upon input of a Concept Unique Identifier (CUI), use the following option:

```
-ty:inp cui
```

To request concept information based upon an entered term name, use the following option:

```
-ty:inp term
```

In addition to specifying an entry term, you can also control the searching criteria for that term name using the following options:

Option	Description
`-wd`	Requests a search for the term name using the UMLS word index
`-nw`	Requests a search for the term name using the UMLS normalized word index
`-ns`	Requests a search for the term name using the UMLS normalized string index
`-am`	Requests a search for the term name using an approximate matching scheme
`-tl`	Requests a search for the term name where the entered term is left-truncated
`-tr`	Requests a search for the term name where the entered term is

right-truncated

3.1.2 Controlling Command-Line Output

The command line interface allows users to specify an output type for the returned data. Both XML output and text output are supported by the interface. To request output of the returned request in simple text format, use the following option:

```
-ty:out text
```

To request output of the returned request in XML, use the following option:

```
-ty:inp xml
```

3.1.3 Restricting Search to a Set of Sources

The default search will look for data in all of the UML sources. To restrict the search to a set of sources, use the following option:

```
-svo <SOURCE_LIST>
```

where `<SOURCE_LIST>` is a comma delineated list of valid sources. (The list of valid sources is available as a separate document and not reproduced here.)

3.2 Requesting Concept Details

Concepts found within the Metathesaurus can be displayed using the command-line interface. The command-line interface window requests the entry of a Unix-like command with options specified for the type of concept to be returned. The window shown below appears when the command-line interface is selected from the advanced search options window. The following sections describe the various options available to users requesting concept data.

3.2.1 Direct Concept-Related Options

The following table lists those options that together describe the complete concept and their effect on the resulting return data.

Option	Description
-cst	Requests return of the semantic type for the concept.
-def	Requests return of the definition for the concept.
-anc	Requests return of the ancestors for the concept.
-par	Requests return of the parents for the concept
-chd	Requests return of the children of the concept
-sib	Requests return of the siblings of the concept

3.2.2 Concept's Relation Options

The relationships for a concept can be categorized into 4 categories: narrower concepts, broader concepts, similar concepts, or other concepts.

To request the list of narrower concepts for the entered concept request, use the option

```
-cr:n
```

To request the list of broader concepts for the entered concept request, use the option

```
-cr:b
```

To request the list of similar concepts for the entered concept request, use the option

```
-cr:l
```

To request the list of concepts related by some other characterization (excluding narrower, broader, or similar) for the entered concept request, use the option

```
-cr:o
```

3.2.3 Associated Expressions Option

The associated expressions (the expressions related to the query term in another vocabulary) for a concept can be returned using the option:

```
-atx
```

These expressions are given as Boolean formulas and are intended as guidelines for searching that meaning in the other vocabulary.

3.2.4 Co-occurring Concepts Option

The co-occurring concepts for a concept can be returned using the option:

```
-cot
```

For a given term, this query retrieves co-occurring terms with the frequency of their co-occurrence in MEDLINE citation records or in AI/RHEUM records. For MeSH, co-occurrence information is computed only for "starred" MeSH terms, those that are deemed to be the major points of articles.

3.2.4.1 Co-occurring Qualified Terms Option

For a given MeSH qualifier, this query can be configured to retrieve the MeSH terms and the frequency with which they co-occur with a particular qualifier in MEDLINE citation records. To specify a qualifier for which to search, use the following option instead of the generic -cot option.

```
-cot:q <QUALIFIER>
```

where `<QUALIFIER>` is one of the valid cooccurrence qualifiers described in the UMLS documentation (http://www.nlm.nih.gov/research/umls/METAB2.HTML).

3.2.4.2 Co-occurring Qualifiers Option

For a given MeSH qualifier, this query can be configured to retrieve the MeSH qualifiers and the frequency with which they co-occur with the given MeSH term in MEDLINE citation records. Co-occurrence information is computed only for "starred" MeSH terms, those that are deemed to be the major points of articles. Given two MeSH terms, this query retrieves the qualifiers for the first term whenever it co-occurred with the second term. To specify this option, use the following option instead of the generic -cot option.

```
-cot:qb <TERM_1> <TERM_2>
```

where `<TERM_1>` and `<TERM_2>` are placed in double quotes.

The request for co-occurring qualifiers can also be displayed as grouped by their semantic types. For a given term, the request returns the co-occurring terms grouped under their semantic types with their

corresponding frequencies. Type frequency indicates the number of co-occurring terms having that semantic type, while term frequency indicates the number of times the query term is found with the co-occurring term. To request this option, using the command option:

```
-cot:st <SEMANTIC_TYPE>
```

where `<SEMANTIC_TYPE>` is the requested semantic type.

3.2.5 Locator Option

The locator query provides information about databases in which a concept appears. The frequency of the occurrence of the concept in the database is provided if available. To specify this option, use the following option:

```
-lo
```

3.3 Requesting Term Details

Terms found within the Metathesaurus can be displayed using the command-line interface. The command-line interface window requests the entry of a Unix-like command with options specified for the term to be returned. The window shown below appears when the command-line interface is selected from the advanced search options window. The following sections describe the various options available to users requesting term data. To request the return of term information, use the option:

```
-t
```

Given a term, the 'term information' query provides the name of the term, the Metathesaurus lexical unique identifier, and the list of sources in which that term appears. For each such source, the source abbreviation is listed first, then the term type in that particular source (see Appendix B of the UMLS Documentation http://www.nlm.nih.gov/research/umls/METAB3.HTML) for the expected term type values), and then the unique identifier of the term in its source vocabulary.

3.3.1 Synonyms/Term Variants Option

The synonyms or term variants for an entered term are requested using the following option:

```
-tv
```

For the given term, this query retrieves the lexical variations of the term, the type of variation expressed, e.g., singular, and the source vocabulary of the variant.

3.3.2 MeSH Entry Terms Option

The MeSH entry terms for an entered term are requested using the following option:

```
-et
```

For the given term, this query retrieves the MeSH entry terms for that heading. The MeSH entry terms are those that are printed in Index Medicus with the label "EP" and not labeled "EN".

3.3.3 Allowable Topical Qualifiers Option

The allowable topical qualifiers for an entered term are requested using the following option:

```
-aq
```

For the given term, this query retrieves the potential qualifiers for that MeSH heading in MEDLINE citation records.

3.3.4 Qualified Terms Option

The qualified terms for an entered term are requested using the following option:

```
-qb
```

For a given allowable qualifier, this query returns a list of terms that may be qualified by that qualifier.

3.3.5 String Attributes Option

The string attributes for an entered term are requested using the following option:

```
-sat
```

For a given allowable term, when selecting the string attributes, you may restrict the query to all string attributes or the attributes described in the following table.

Option	Description	Option	Description
-am	Ambiguous string indicator	-med	Medline Postings
-an	Annotation	-mmr	MeSH revision date
-aq	Allowable topical qualifier for	-mn	MeSH Tree Number
-cx	Consider Also Note	-da	Date of entry
-th	Thesaurus ID	-mr	Major revision date
-dc	Descriptor class	-na	Neuronames Abbreviation
-de	Descriptor Entry Version	-nat	Neuronames Anatomy Type
-dq	Date Qualifier Established	-no	Note
-ds	Descriptor Sort Version	-nh	Non-Human Flag
-dx	Date major descriptor established	-nsr	Neuronames Species Restriction
-dy	Date minor descriptor established	-ol	Online Note
-ec	Entry combination	-pa	Pharmacologic Action
-ev	Entry term abbreviation	-pda	PDQ Short Name or Abbreviation
-ez	SNOMED International Enzyme Commission Number	-pi	Previous Indexing
-fr	Frequency	-pm	Public MeSH note
-gc	Geographic Area Code	-px	Pre-Explosion
-gm	Grateful Med Note	-qa	Topical Qualifier Abbreviation
-hn	History note	-qb	Qualified By
-ii	Index information	-qe	Qualifier Entry Version
-la	Language Code	-qg	Qualifier Usage
-lat	Language of Term	-qs	Qualifier Sort Version
-lo	Locator	-qt	Qualifier Type
-lt	Lexical Tag	-qu	Qualifier Used
-lu	Language Usage	-qx	Qualifier Cross reference
-m66	M66 Postings	-rn	Registry Number
-m72	M72 Postings	-rr	Related Registry Number
-m75	M75 Postings	-sb	SNOMED International subset indicator
-m77	M77 Postings	-sct	Syntactic Category

-m80	M80 Postings	-sic	SNOMED ICD-9-CM Reference
-m83	M83 Postings	-smx	SNOMED Multi-axial coding
-m85	M85 Postings	-sos	Scope statement
-m86	M86 Postings	-src	Literature source of chemical name
-m89	M89 Postings	-st	Concept Attribute Status
-mda	MeSH date of entry		

3.4 Viewing Raw Metathesaurus Database Tables

The ASCII relational records used to populate the backend database of the UMLSKS can be returned to the user via the command-line interface. To request the return of the raw table data, use the following option:

```
-raw <TABLE_NAME>
```

where <*TABLE_NAME*> is one of the tables shown in the drop down menu as described in Section 2.4.5.

3.5 Requesting the UMLS Release Year

The UMLS release is requested using the following option:

```
-y
```

4 Using the XML Query Facility

The UMLSKS API provides an extensive query facility for users expert in the layout of the UMLS data and the corresponding database tables. The facility utilizes the power of the eXtensible Markup Language (XML) to request the return of any data found within the UMLS data set. A special language was developed for the UMLSKS to allow users to develop an XML query that may be issued to the UMLSKS for processing. This chapter describes this process.

4.1 Retrieving Metathesaurus, Semantic Network, and SPECIALIST Lexicon Data using an XML Query

Retrieving Metathesaurus data using the XML query feature of the UMLSKS requires an understanding of the UMLS structures, the UMLSKS internal database design and the theory behind querying a relational database. The returned XML document from the query uses Oracle's XML generation facility to generate the XML, so the XML will not look like the XML generated by the other UMLSKS API methods. The XML tags for the columns will be the names as they appear in Oracle (i.e. the field names shown in Section 4.3 will be the tags for the generated XML). In addition, the XML returned using this scheme of data retrieval cannot be used in conjunction with the object model classes.

Once *Perform XML Query* is selected from the *Metathesaurus Resources* list, the following window will be displayed to the user.

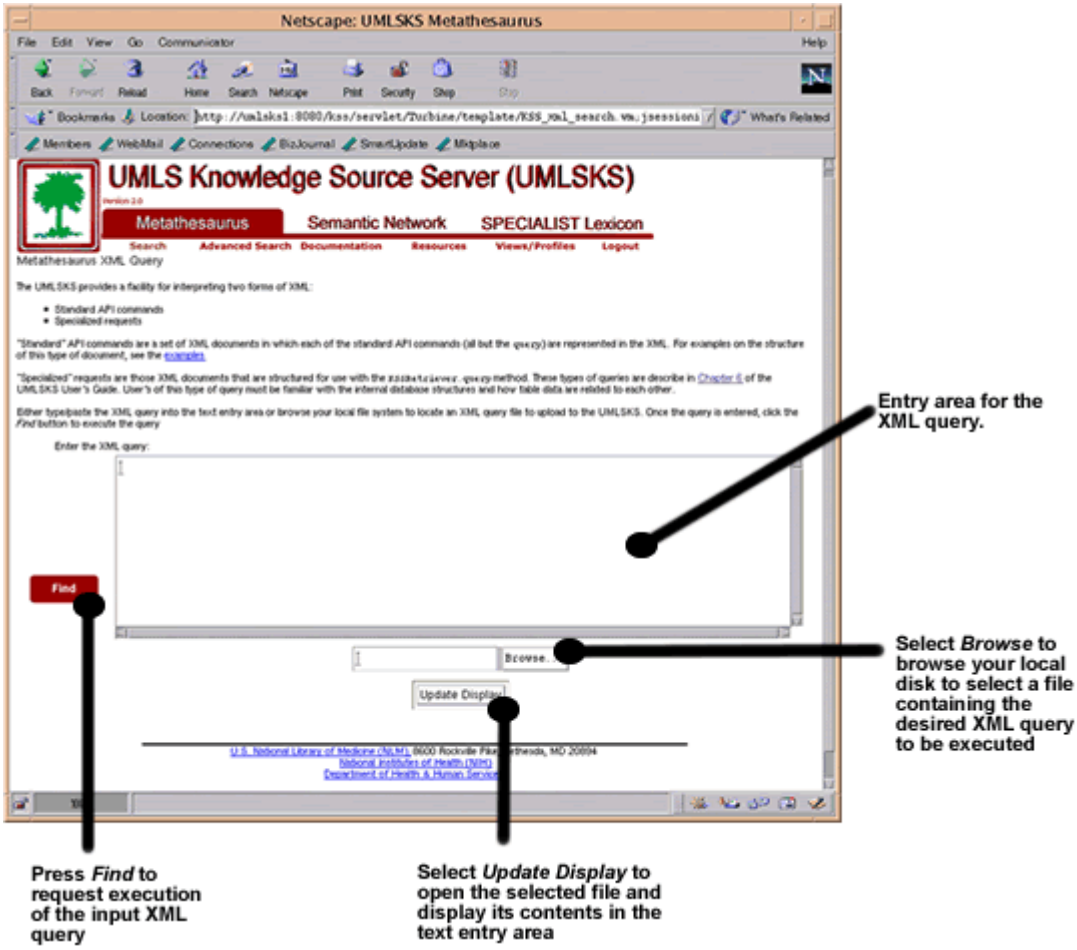


Figure 38 – Requesting XML Query Execution

Users may enter text directly into the text entry area or may browse the local file system and upload a file containing the desired XML query. Once entered/displayed, pressing the *Find* button will execute the XML query and display the resulting XML to the user.

4.2 Structure of an XML Query

A UMLSKS query is an XML document that describes a request for some subset of data from the Metathesaurus. The default namespace is used for all elements in the XML definition. The elements within a query document are described below using regular expression syntax. Sample queries are also provided in Section 0.

Formatting of the regular expressions describing the XML grammar following these conventions:

- Items in **bold** are ASCII characters that should appear exactly as shown in bold.
- The '|' character indicates an 'or' relationship. For example, the *fieldSetGroup* item can be either a *fieldSet* followed by a *fieldSetGroup*, or a single *fieldSet*
- ε denotes an empty value (i.e. may not be present)

```
query -> <query version=1.0> release rowset row distinct fields
               constraints ordering </query>

release -> <release/>string<release/>  | ε
rowset  -> <rowset>string</rowset>
row  -> <row>string</row>
distinct -> <distinct/> | ε
fields -> <fields> allFields </fields> | <fields> fieldSetGroup </fields>
allFields -> <all> tableSpec </all>
tableSpec -> tableNameSpec tableSpec | tableNameSpec
fieldSetGroup -> fieldSet fieldSetGroup | fieldSet
fieldSet -> <fieldSet> tableNameSpec fieldPart </fieldSet>
tableNameSpec -> <table> tableName </table>
tableName -> mrcon | mrso | mrdef | mrsty |
           mrrel | mratx | mrcxt | mrcoc | mrsat | mrlo |
           mrxns_eng | mrxnw_eng
fieldPart -> <fieldNames> fieldNameSpec </fieldNames> fieldPart |
           <fieldNames> fieldNameSpec </fieldNames>

constraints -> <constraints> constraintList </constraints> | ε
constraintList -> relation constraintList  |
         constraint constraintList | relation | constraint
relation -> <relation> relation_op constraintList </relation>
relation_op -> <operator> op </operator>
op -> AND | OR | NOT

constraint -> <constraint> const </constraint>
const -> lhs | lhs const_op rhs
lhs -> <lhs> fieldList </lhs>  | <lhs> field <lhs>
fieldList -> <fieldList> fieldSpec </fieldList>
fieldSpec -> fieldNameSpec fieldSpec | fieldNameSpec
fieldNameSpec -> <name> fieldNameSpec </name>
field -> <field> tableName.fieldName </field>
fieldName -> Field name for the appropriate table as described in the following sections

const_op -> <op> operator </op> | normStr | normWord | word | like
operator -> oneof | noneof | = | >= | <= | > | < | !=
normStr -> <normStr> string </normStr>
normWord -> <normWord> string </normWord>
word -> <word> string </word>
like -> <like> string </like>

rhs -> <rhs> query </rhs> | <rhs> stringList </rhs> |
         <rhs> stringSpec </rhs> | <rhs> field </rhs>
stringList -> <stringList> strings </stringList>
strings -> stringVal strings | stringVal
stringVal -> <name> string </name>
stringSpec -> <string> string</string>
string -> set of characters that is readable by an XML parser (i.e. has the < described as &lt; and the '>' described
        as &gt;)

ordering -> <orderBy> fieldSpec</orderBy>
```

4.3 Metathesaurus and Semantic Network Table Contents

The following sections describe the tables and their field names that may be used as the <*tableName*> and <*fieldName*> values described in the grammar.

4.3.1 Table 'mrcon'

The 'mrcon' table for UMLS releases prior to **2004AA** contains the following fields:

Field Name	Description
CUI	Concept Unique Identifier
LAT	Language
TS	Term Status
LUI	Lexical Unique Identifier
STT	String Type
SUI	String Unique Identifier
STR	String Value for the concept
LRL	Least Restriction Level
STR2	Normalized version of the concept's string value in all uppercase

4.3.2 Table 'mrso'

The 'mrso' table for UMLS releases prior to **2004AA** contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
LUI	Lexical Unique Identifier
SUI	String Unique Identifier
SAB	Source Abbreviation
TTY	Term Type
SCD	Unique identifier or code for string in that source
SRL	Source Restriction Level

4.3.3 Table 'mrdef'

The 'mrdef' table contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
SAB	Source abbreviation
DEF	Definition

4.3.4 Table 'mrsty'

The 'mrsty' table contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
TUI	Unique Identifier of the Semantic type
STY	Semantic type

4.3.5 Table 'mrrel'

The 'mrrel' table for UMLS releases **prior** to 2004AA contains the following fields that may be specified as

Field Name	Description
CUI1	Concept Unique Identifier 1
REL	Relationship of second concept to the first
CUI2	Concept Unique Identifier 2
RELA	Relationship attribute
SAB	Abbreviation of the source of the relationship
SL	Source of relationship labels
MG	Machine-generated and unverified indicator

The 'mrrel' table for UMLS release **2004AA** and later releases contains the following fields that may be specified as

Field Name	Description
CUI1	Concept Unique Identifier 1
AUI1	Atomic unique identifier 1
STYPE1	Source asserted type for identifier 1
REL	Relationship of second concept to the first
CUI2	Concept Unique Identifier 2
AUI2	Atomic unique identifier 2
STYPE2	Source asserted type for identifier 2
RELA	Relationship attribute
RUI	Relationship unique identifier
SRUI	Source attributed relationship identifier
SAB	Abbreviation of the source of the relationship
SL	Source of relationship labels
RG	Relationship Group
DIR	Source asserted directionality flag
SUPPRESS	Suppressible flag
CVF	Content view flag

4.3.6 Table 'mratx'

The 'mratx' table for UMLS releases prior to **2004AA** contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
SAB	Abbreviation of source of terms in expression
REL	Relationship of meaning of expression to main concept
ATX	Associated expression

4.3.7 Table 'mrcxt'

The 'mrcxt' table for UMLS releases prior to **2004AA** contains the following fields that may be specified as

Field Name	Description

CUI	Concept Unique Identifier
SUI	String Unique Identifier
SAB	Source abbreviation
SCD	Unique identifier or code for string in that source
CXN	The context number (to distinguish multiple contexts in the same source with the same SUI)
CXL	Context member label (i.e. ANC for ancestor of this concept, CCP for concept, SIB for sibling of this concept, CHD for child of this concept).
RNK	For rows with a CXL value of ANC, the rank of the ancestors (e.g. a value of 1 denotes the most remote ancestor in the hierarchy)
CXS	String for context member
CUI2	Unique concept identifier of context member
HCD	Hierarchical number or code of context member in this source
RELA	Relationship attribute providing further categorization of the CXL
XC	A plus (+) sign indicates that the CUI2 for this row has children in this context

The 'mrcxt' table for UMLS release **2004AA** and later releases contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
SUI	String Unique Identifier
AUI1	Atomic unique identifier for this atom
SAB	Source abbreviation
CODE	Unique identifier or code for string in that source
CXN	The context number (to distinguish multiple contexts in the same source with the same SUI)
CXL	Context member label (i.e. ANC for ancestor of this concept, CCP for concept, SIB for sibling of this concept, CHD for child of this concept).
RNK	For rows with a CXL value of ANC, the rank of the ancestors (e.g. a value of 1 denotes the most remote ancestor in the hierarchy)
CXS	String for context member
CUI2	Unique concept identifier of context member
AUI2	Atomic unique identifier for second atom
HCD	Hierarchical number or code of context member in this source
RELA	Relationship attribute providing further categorization of the CXL
XC	A plus (+) sign indicates that the CUI2 for this row has children in this context
CVF	Content view flag

4.3.8 Table 'mrcoc'

The 'mrcoc' table for UMLS releases prior to **2004AA** contains the following fields that may be specified as

Field Name	Description
CUI1	Concept Unique Identifier 1
CUI2	Concept Unique Identifier 2
SOC	Abbreviation of the source of the co-occurrence information
COT	Co-occurrence type
COF	Co-occurrence frequency
COA	Attributes of the co-occurrence

The 'mrcoc' table for UMLS release **2004AA** and later releases contains the following fields that may be specified as

Field Name	Description
CUI1	Concept Unique Identifier 1
AUI1	Atomic unique identifier 2
CUI2	Concept Unique Identifier 2
AUI2	Atomic unique identifier 2
SAB	Abbreviation of the source of the co-occurrence information
COT	Co-occurrence type
COF	Co-occurrence frequency
COA	Attributes of the co-occurrence
CVF	Content View Flag

4.3.9 Table 'mrsat'

The 'mrsat' table for UMLS releases prior to **2004AA** contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
LUI	Term Unique Identifier
SUI	String Unique Identifier
SCD	Unique identifier or code for entry in the source of the attribute
ATN	Attribute name
SAB	Abbreviation of the source of the attribute
ATV	Attribute value

The 'mrsat' table for UMLS release **2004AA** and later releases contains the following fields that may be specified as

Field Name	Description

CUI	Concept Unique Identifier
LUI	Term Unique Identifier
SUI	String Unique Identifier
UI	Atom or Relationship identifier
STYPE	Source asserted type for the identifier in CODE
CODE	Unique identifier or code for entry in the source of the attribute
ATUI	Attribute unique identifier
SATUI	Source asserted attribute identifier
ATN	Attribute name
SAB	Abbreviation of the source of the attribute
ATV	Attribute value
SUPPRESS	Suppressible flag
CVF	Content View Flag

4.3.10 Table 'mrlo'

The 'mrlo' table for UMLS releases prior to **2004AA** contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
ISN	Name of information source or database in which concept appears
FR	Frequency count of number of occurrences of concept in the information source (optional)
UN	Meaning of the frequency
SUI	Unique identifier of the string if name used in information source appears in MRCON
SNA	Actual name that occurs in the information source if not otherwise present in the Metathesaurus
SOUI	Unique identifier of record in which the concept appears in the source

The 'mrlo' table for UMLS release **2004AA** and later releases contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
AUI	Atomic unique identifier
ISN	Name of information source or database in which concept appears
FR	Frequency count of number of occurrences of concept in the information source (optional)
UN	Meaning of the frequency
SUI	Unique identifier of the string if name used in information source appears in MRCON
SNA	Actual name that occurs in the information source if not otherwise present in the Metathesaurus

SOUI	Unique identifier of record in which the concept appears in the source
CVF	Content View Flag

4.3.11 Table 'mrxns_eng'

The 'mrxns_eng' table contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
LUI	Lexical Unique Identifier
SUI	Unique identifier of the string if name used in information source appears in MRCON
LAT	Language
NSTR	Normalized string value
NSTR2	Normalized string value with spaces removed and all letters capitalized

4.3.12 Table 'mrxnw_eng'

The 'mrxnw_eng' table contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
LUI	Lexical Unique Identifier
SUI	Unique identifier of the string if name used in information source appears in MRCON
LAT	Language
NWD	Normalized word value

4.3.13 Table 'mrxw_eng'

The 'mrxw_eng' table contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
LUI	Lexical Unique Identifier
SUI	Unique identifier of the string if name used in information source appears in MRCON
LAT	Language
WD	Normalized word value

4.3.14 Table 'mrcui'

The 'mrcui' table for UMLS releases prior to **2004AA** contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
YEAR	UMLS Release
FLAG	Indicator for whether the concept was deleted (value of DEL) or merged (value of SY)

MCUI	When the flag indicates merge, this contains the CUI into which the CUI was merged

The 'mrcui' table for UMLS release **2004AA** and later releases contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
YEAR	UMLS Release
REL	Relationship
RELA	Relationship attribute
MAPREASON	Reason for mapping
CUI2	Unique identifier for the second concept
MAPIN	Indicates presence in subset

4.3.15 Table 'mrsab'

The 'mrsab' table contains the following fields that may be specified as

Field Name	Description
VCUI	CUI for the versioned source
RCUI	CUI for the root source
SAB	Abbreviation of the root source
STR	Official name for the source
STR2	Abbreviation of the versioned source
SF	Source family
VER	Version indicator
MSTART	Date at which the source first appeared in the Metathesaurus
MEND	Date at which this source disappeared from the Metathesaurus
IMETA	Metathesaurus release in which this source first appeared
RMETA	Metathesaurus release in which this source disappeared
SLC	Contact information for obtaining the source's license
SCC	Contact information for information about the source's content
SRL	Source restriction level
TFR	Number of terms for this source in MRCON/MRSO
CFR	Number of CUIs associated with this source
CXTY	The type of context for this source from the UMLS documentation
TTYL	Term-type list from the source
ATNL	Attribute name list from the source
LAT	Language indicator
CENC	Character encoding value
CURVER	Current Version Flag (Y/N)
SABIN	Source in Metamorphosys subsystem flag (Y/N)
SSN	Field added in **2004AA** and later UMLS releases that indicates the source's short name

4.3.16 Table 'mshqual'

The 'mshqual' table for UMLS releases prior to **2004AA** contains the following fields that may be specified as

Field Name	Description
CODE	MeSH Qualifier Code (2-3 characters)
NAME	MeSH Qualifier description

4.3.17 Table 'strattr'

The 'strattr' table for UMLS releases prior to **2004AA** contains the following fields that may be specified as

Field Name	Description
ATN	Attribute name
ATEN	Attribute description

4.3.18 Table 'ttys'

The 'ttys' table for UMLS releases prior to **2004AA** contains the following fields that may be specified as

Field Name	Description
TTY	Term type
STR	Description of the term type

4.3.19 Table 'cots'

The 'cots' table for UMLS releases prior to **2004AA** contains the following fields that may be specified as

Field Name	Description
COT	Cooccurrence type (2-3 characters)
STR	Cooccurrence type description

4.3.20 Table 'stts'

The 'stts' table for UMLS releases prior to **2004AA** contains the following fields that may be specified as

Field Name	Description
STT	String type
STR	Description of the string type

4.3.21 Table 'rels'

The 'rels' table for UMLS releases prior to **2004AA** contains the following fields that may be specified as

Field Name	Description
REL	Relation type
STR	Description of the relation type

4.3.22 Table 'srdef'

The 'srdef' table contains the following fields that may be specified as

Field Name	Description
RT	Record type (STY=Semantic Type or RL=Relation)
UI	Unique identifier of the semantic type or semantic relation
NAME	Name of the semantic type or semantic relation
TN	Tree number of the semantic type or semantic relation
DEF	Definition of the semantic type or relation
EX	Examples of Metathesaurus concepts with this semantic type (STY records only)
UN	Usage note for semantic type assignment (STY records only)
NH	Semantic type and its descendants allow the non-human flag (STY records only)
ABBREV	Abbreviation of the semantic relation name (RL records only)
RIN	Inverse of the semantic relation (RL records only)

4.3.23 Table 'srgrp'

The 'srgrp' table contains the following fields that may be specified as

Field Name	Description
ABBREV	Abbreviation for the semantic group
GRP	Group name
STNAME	Semantic type name belonging to the semantic group
STUI	Unique identifier of the semantic type belonging to the semantic group

4.3.24 Table 'srstr'

The 'srstr' table contains the following fields that may be specified as

Field Name	Description
SRLEFT	Argument 1 (name of a semantic type or semantic relation)
REL	Relation ("isa" or the name of a non-hierarchical semantic relation)
SRRIGHT	Argument 2 (name of a semantic type or semantic relation); if this field is blank this means that the semantic type or semantic relation is one of the top nodes of the Network.
LS	Link Status (D = Defined for the arguments and its children; B = Blocked; DNI = Defined but Not Inherited by the children of the arguments) The relations expressed in this table are binary relations and the arguments are ordered pairs. The relations are stated only for the top-most node of the "isa" hierarchy of the semantic types to which they may apply.

4.3.25 Table 'srstre1'

The 'srstre1' table contains the following fields that may be specified as

Field Name	Description
LTUI	Argument 1 semantic type unique identifier
REL	Semantic relation unique identifier
RTUI	Argument 2 semantic type unique identifier

4.3.26 Table 'srstre2'

The 'srstre2' table contains the following fields that may be specified as

Field Name	Description
SRLEFT	Argument 1 semantic type name
REL	Semantic relation name
SRRIGHT	Argument 2 semantic type name

4.3.27 Table 'mrconso'

The 'mrconso' table for UMLS release **2004AA** and later contains the following fields that may be specified as

Field Name	Description
CUI	Concept Unique Identifier
LAT	Language
TS	Term Status
LUI	Lexical Unique Identifier
STT	String Type
SUI	String Unique Identifier
ISPREF	Indicates preferred term for the language indicated for the row
AUI	Atomic unique identifier
SAUI	Source asserted unique identifier
SCUI	
SDUI	
SAB	Source abbreviation
TTY	Term type
CODE	Unique identifier or code for the string in the source
STR	String Value for the concept
SRL	Least Restriction Level
SUPPRESS	Suppressible flag
CVF	Content View Flag
STR2	Normalized version of the concept's string value in all uppercase

4.3.28 Table 'mrdoc'

The 'mrdoc' table contains the following fields that may be specified as

Field Name	Description
TYPE	Type of the meta data
MKEY	Key to the type of meta data
SUBKEY	Key to the abbreviation for the entry of the MKEY type
VALUE	Value for the SUBKEY

4.3.29 Table 'mrhier'

The 'mrhier' table contains the following fields that may be specified as

Field Name	Description
CUI	Concept unique identifier
AUI	Atomic unique identifier
CXN	
PAUI	
SAB	
RELA	
PTR	
HCD	
CVF	Content view flag

4.3.30 Table 'mrhist'

The 'mrhier' table contains the following fields that may be specified as

Field Name	Description
CUI	Concept unique identifier
UI	
SAB	
VER	
CHANGETYPE	
CHANGEKEY	
CHANGEVAL	
REASON	

4.3.31 Table 'mrmap'

The 'mrmap' table contains the following fields that may be specified as

Field Name	Description
MAPSETCUI	
MAPSETSAB	
MAPSUBSETID	
MAPRANK	
FROMID	
FROMSID	
FROMEXPR	
FROMTYPE	
FROMRULE	
FROMRES	
REL	
RELA	
TOID	
TOSID	
TOEXPR	
TOTYPE	
TORULE	

TORES
MAPRULE
MAPTYPE
MAPATN
MAPATV
CVF Content View Flag

4.4 XML Query Example

A number of sample queries are provided with the API delivery, including queries for all of the convenience API functions describe previously. The XML query that returns the details of a concept, similar to the `getConcept` API method, is described here.

The following example query requests the return of all the fields in the "mrcon", "mrso", "mrsty", and "mrcxt" tables where the concept unique identifier is 'C0024109'.

```
<?xml version="1.0"?>
<query xmlns="http://umlsks4.nlm.nih.gov"
      xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-instance"
      version="1.0">

   <release>2002</release>
   <distinct/>

   <fields>
      <all>
         <table>mrcon</table>
         <table>mrso</table>
         <table>mrsty</table>
         <table>mrcxt</table>
      </all>
   </fields>

   <constraints>
      <relation>
         <operator>AND</operator>
         <constraint>
               <lhs><field>mrcon.cui</field></lhs>
               <op>=</op>
               <rhs><string>C0024109</string></rhs>
         </constraint>
         <constraint>
               <lhs><field>mrcon.ts</field></lhs>
               <op>!=</op>
               <rhs><string>s</string></rhs>
         </constraint>
         <constraint>
               <lhs><field>mrcon.cui</field></lhs>
               <op>=</op>
               <rhs><field>mrso.cui</field></rhs>
         </constraint>
         <constraint>
               <lhs><field>mrcon.lui</field></lhs>
               <op>=</op>
               <rhs><field>mrso.lui</field></rhs>
         </constraint>
```

```
            <constraint>
                  <lhs><field>mrcon.sui</field></lhs>
                  <op>=</op>
                  <rhs><field>mrso.sui</field></rhs>
            </constraint>
            <constraint>
                  <lhs><field>mrcon.cui</field></lhs>
                  <op>=</op>
                  <rhs><field>mrsty.cui</field></rhs>
            </constraint>
            <constraint>
                  <lhs><field>mrcon.cui</field></lhs>
                  <op>=</op>
                  <rhs><field>mrcxt.cui</field></rhs>
            </constraint>
         </relation>
      </constraints>

      <orderBy>
         <name>mrcon.cui</name>
         <name>mrcon.sui</name>
         <name>mrso.sab</name>
         <name>mrcxt.scd</name>
         <name>mrcxt.cxn</name>
         <name>mrcxt.rnk</name>
      </orderBy>
   </query>
```