INFORMATION SERVER

Verity's application suite includes the Information Server, which provides the foundation for advanced search technology. The Information Server creates and manages indexed information about your documents, and dynamically accesses these indexes for high-speed search and retrieval of relevant information.

The Information Server offers:

- An easy-to-use graphical administrator with pre-defined query forms and results
- Advanced application customization through SEARCHScript, an HTML-based language with user defined variables and logic
- Indexing support for HTML, WYSIWYG, ASCII, and PDF document formats
- Access to existing Verity collections

THE INFORMATION SERVER uses a modular streaming architecture, specifically designed to accommodate integration with new search modules from Verity or third-parties. All components are integrated within this open architecture, which consolidates information retrieval services and simplifies administration. Additional application components can be added as needs evolve.

THE INFORMATION SERVER also includes client products designed to help users find information quickly. The advanced filtering components provided by Verity's KeyView filters are included in the application suite and allow users to view documents in their native format without actually launching the applications that created them.

EXPLORING THE INFORMATION SERVER

The Information Server is the foundation of your search application. It provides access to the components that allow you to enable search on your documents. These components include collections that contain information about your documents and the search language, which allows you to query the collections to extract relevant information.

The Role of Collections

The term collection represents a group of documents that include search elements, which facilitate rapid retrieval of documents relevant to a query:

- Indexes are optimized for searching.
- References actual documents locations
- Stores field values or other document attributes that allow you to search standard document information like title or date

A collection is a series of indexes, which store data about documents.

- The documents themselves are never stored in a collection.
- Collections are configured through style files, which define the guidelines governing the creation of the collection. The style becomes a physical part of the collection.
- Multiple document formats may exist in one collection.

- Collections allow for querying and updating at the same time.
- One command can create a collection:
- Collections are fault tolerant. If new work fails, the original collection will be unharmed.
- Collections are self-administering. If a collection goes down, it will attempt to repair itself.
- Collections are portable and include real-time functionality for internal management.

The Role of Query Language

Verity's query language consists of a robust set of terms that precisely target information based on:

- Content within a document
- Content within zones in a document
- Content within fields defined for a document

In addition to the standard query language, Verity products include topics, which allow you to extend the query language to include your terminology for more precise results.

- Topics capture areas of general interest and can be shared globally throughout the enterprise
- Verity products come with utilities to turn text topics into knowledgebase objects

COMPONENTS WORKING TOGETHER

Your application consists of Information Server software, collections you create with indexing tools, templates you can customize to simplify searching and using information, and a configuration file that keeps track of everything.

PRODUCT FEATURES

Features are organized by major categories of functionality.

- Server features include integration into your HTTP server, a graphical administration module that allows you to configure your server and tune performance.

- Indexing features include a graphical Indexing Manager and Quick Start that allows you to easily submit indexing tasks, capturing information about each task in a log. There is also a Collection Manager that allows you to import existing collections, enable or disable collections for the server and obtain information about your collections at any time.

- Search features include a Forms Wizard to help you select how you want to enter queries and have results presented and the default templates to be used. You have Verity's robust query language for searching and SEARCHScript templates for viewing, filtering and presenting results.

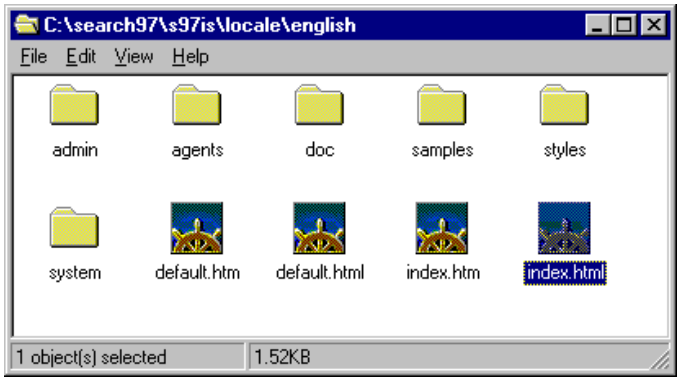
INFORMATION SERVERS

When you install the Information Server, settings are updated in the HTTP server you selected. After you install the software you will need to make a couple of edits to web server configuration files. These updates are detailed in the installation instructions, which you will find in the Additional Materials section of this workbook.

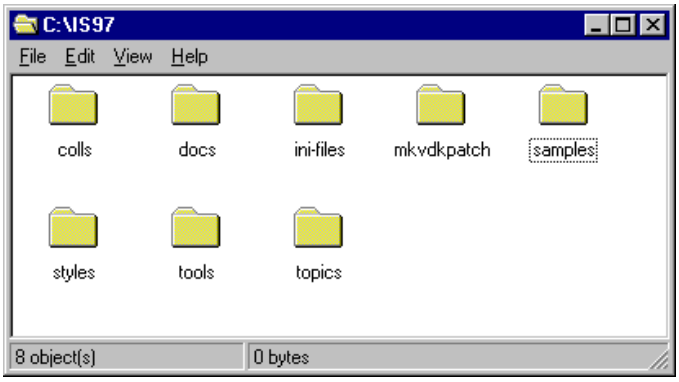
When you "Apply" the changes you will notice that several directory mappings have been automatically created for you – enabling use of the many forms, images, and templates in the application.

BROWSING THE ENVIRONMENT

You can review the files in the application environment by reviewing the files on the file system.



For class, we have provided additional samples in a special directory called IS97. It is a good idea to become familiar with this environment as well since it includes many tools for building search forms and building collections.



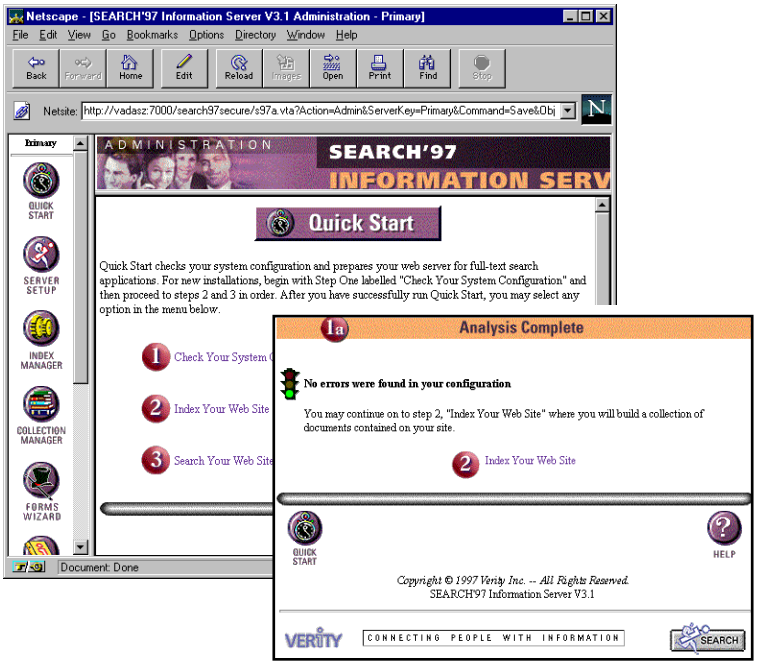
USING THE QUICK START

If you are using the Information Server for the first time, you should begin with the Quick Start. This is the quickest and easiest way to search-enable your site. In three simple steps, the Quick Start prepares your Internet site for searches. After you have successfully run the Quick Start, you can take advantage of advanced features in the Server Setup, Indexing Manager, and Collection Manager components.

The first step in the Quick Start process is to check your system configuration. This analysis performs a check of the configuration file (called inetsrch.ini and located in the bin directory), which will store information about paths to executables, template files, and other server components. If you have just installed your software you should see this analysis is quickly completed and returns the message that "No errors were found in your configuration." If it does not, there may have been a problem with the installation. If these problems do occur, the best thing to do is uninstall the product and then go through the installation process once more. Be sure you have adequate disk space available and that you have checked the port and general

server information for your web server so you have all the information you need during the installation.

The Quick Start is as Easy as 1-2-3



CREATING A COLLECTION

Before you can index your web site, you will need to create an empty collection to receive the document information. This collection gets added to a list of collections, which you can select from, as the destination for new documents. Generally you will set up more than one collection at your site. Because documents often change, you may issue "re-indexing" commands for some of the collections you create. If your data is unchanging, for example historical data, you will not need to re-index the collection because any internal maintenance needed by the collection is handled for you programmatically.

INDEXING YOUR SITE

Once your collection has been added, you are ready to index your web site.

- You will enter the URL you wish to index and then select your destination
- You can add proxy information as appropriate for your site

The Indexing Manager allows you to index your own or other web sites, and to review and manage those indexing tasks. In addition to the Indexing Manager, the Information Server ships with two collection building (command-line) utilities (VSPIDER and MKVDK) to help you build your collections on different types of documents. We will review these utilities a bit later in the course.

MONITORING INDEXING STATUS

Once the indexing task is submitted, activities are captured in a log file that was automatically created for you. When the indexing is complete, you may notice that some documents were skipped. This can be a little confusing, but there are several good reasons for skipping documents. The most common reason for skipping documents is that they are already in the collection. As you know, many webs sites point at the same documents over and over. Jump from the home page to marketing and it will have a link back to the home page. The Information Server is smart enough to know when it already added a particular document to the collection. It verifies that the date and size are the same before assuming that it is the same document.

When you point at the beginning URL for your indexing process, you make a lot of assumptions about the level of access that your indexing process will have to the documents. Here is another reason why documents may be skipped. You simply don't have access (or more exactly your indexing process doesn't have the right access) and it gets an error as it tries to access the page. And, of course, missing documents are skipped.

There are also limits that are set on where the indexing is to occur. The Information Server restricts the spider to the host or to the domain of the URL you have selected. This means that if a link from Verity's home page goes to a customer's home page, like the Boston Globe, your collection will not automatically go index the whole Globe site. The indexing spider gets all the documents within the Verity domain.

INDEXING MANAGER

Over time, you will probably issue multiple indexing tasks for the various collections you create or update. The Indexing Manager allows you to track these individual tasks and makes it easy for you to resubmit tasks to update your collections with new information. When you reissue a task to index a site you have previously indexed, you can be sure the number of documents skipped will be quite high – unless they changed all their documents since the last time you indexed. So, it is not a bad thing when documents are skipped, it just means your Information Server is not doing any unnecessary work.

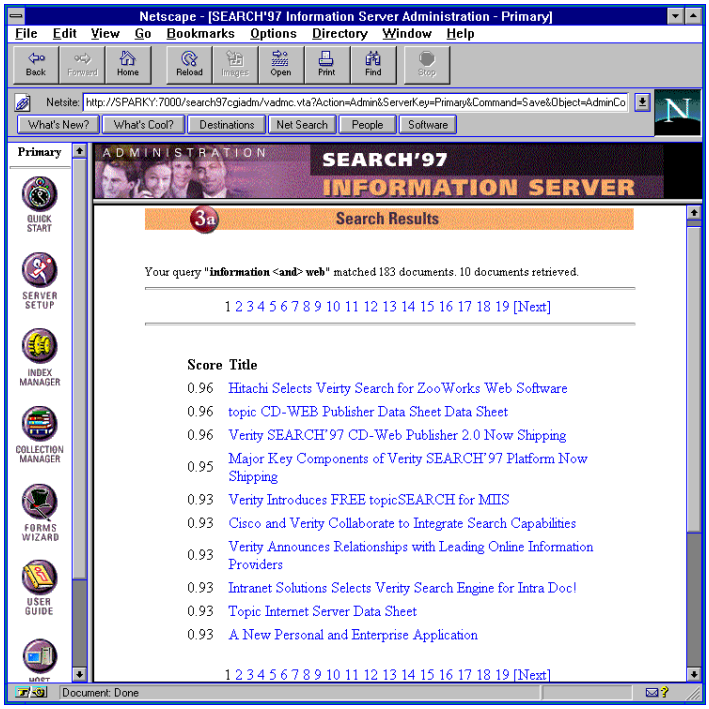
SEARCHING YOUR SITE

The Information Server ships with a default collection containing documentation for the product. Before you begin building collections yourself, you can use this starting collection to test your basic understanding of the search language. How you present information about the collections that you will enable for your users, is a big part of the role you play in designing your application. Verity ships some samples to give you ideas about the kind of functionality you might want to include. Depending upon what you show your users, they can choose to search against all collection (which is the default) or to select a specific one to search from the list of available collections.

VIEWING RESULTS

By default, search results are displayed in a relevance-ranked list. Your job as an administrator is to build the appropriate search forms and results lists to help users select the documents they want to read:

- It is a good idea to bring the query forward to the results list to remind the user what they asked for
- The title, captured automatically during indexing, can be displayed by itself, or along with other information captured during indexing
- In your templates, you can specify that results are to be limited to a specific number per page (10 is the default), and provide additional results list page links at the top and bottom if the count is higher than the number you set for your page



FORMS WIZARD

The Forms Wizard allows you to select other templates, organized as themes, to enhance your searches, results and viewing. To select a theme, click the Forms Wizard button and make the

appropriate choice from the pull down list. It is a good idea to practice with all of these alternatives to get a better sense of your options.

When you click the "apply" button, a special URL is created automatically for you, which you can then place directly in your search form. For example, the URL for the "Simple" theme is **`http://train2:7000/search97cgi/s97_cgi.exe?Action=FormGen&ServerKey=Primary&Template=smplsrcp.hts`** and the URL for the "Power" theme is **`http://train2:7000/search97cgi/s97_cgi.exe?Action=FormGen&ServerKey=Primary&Template=pwersrcp.hts`**

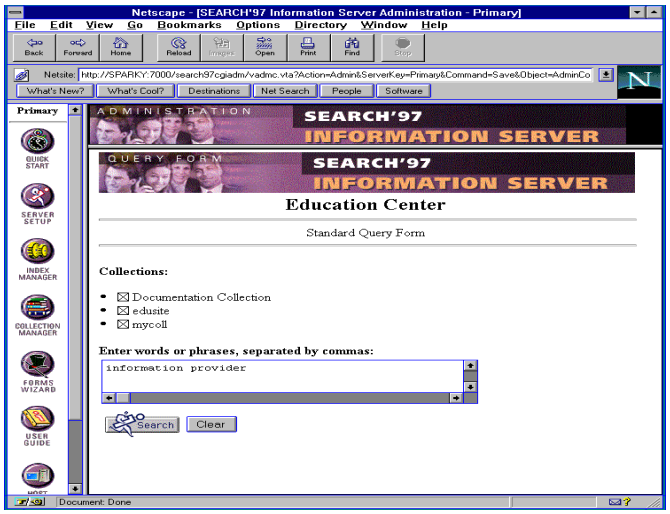
SELECTING QUERY FORMS

The query forms you choose matter a great deal. You have different kinds of users with many different needs. You know best about how your information is organized and how it should be presented to allow people to work with it easily. Here are a few things to consider when planning for your site.

- It is really nice to let people choose what group of documents they want to search but you need to inform them about the differences between collections if these differences are not obvious
- Usually novices need less "search features" than more experienced searchers. Take these needs (and the associated help that goes with them) into consideration when you plan how you will present information.

STANDARD QUERY FORM

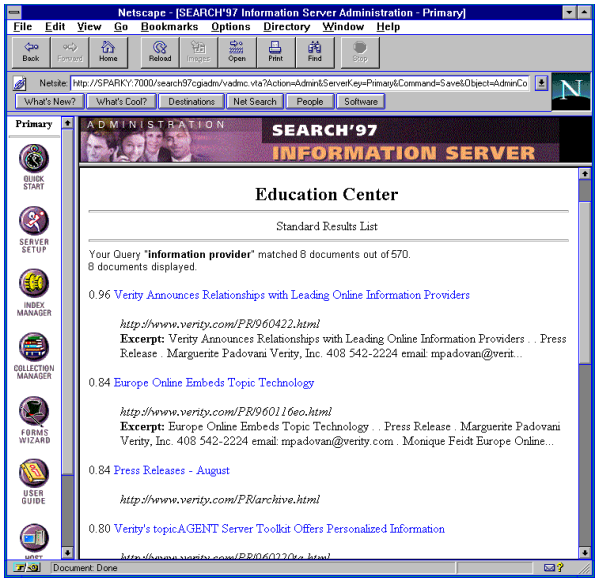
The standard query template allows users to select their document collections, but presents them on the page (rather than from the drop-down list) for user selection:



STANDARD RESULTS LIST

This template provides a good amount of information to help users select the documents they want to read

- Provides details on total found out of the possible total (which is determined by the collections selected)
- Includes a 160 character "snippet" from the document
- Links to a highlighted version of the original HTML document



VIEWING A RESULTS TEMPLATE

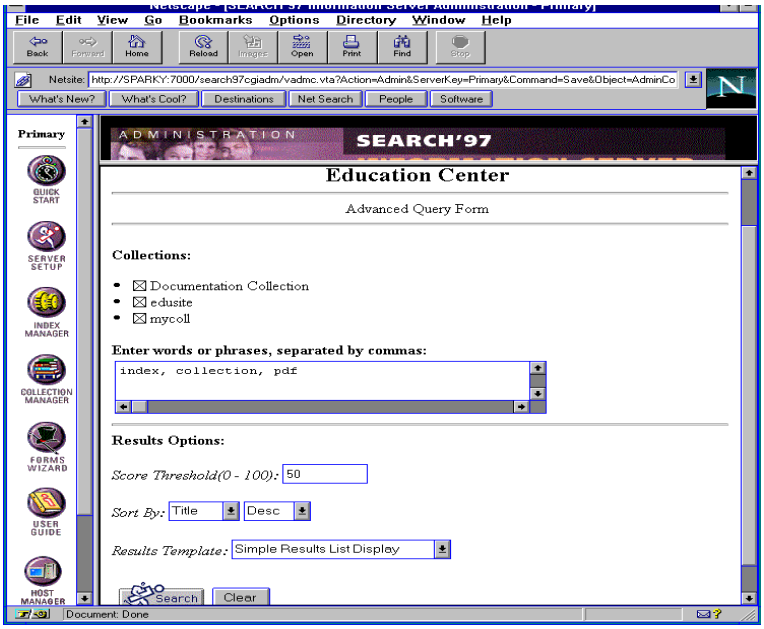
Example of a template using SEARCHScript.

```
<html><head>
<title>Basic Search Results</title></head>
<b><IMG src="$$web.ServerURL/samples/happy.gif"</a> Whoppeeee <br>
Your search matched <b>$$DocsFound</b> of <b>$$DocsSearched</b> documents.<br>
$$DocsCount</b> are presented, with the first part of the document included
to give you a better idea about it's contents.
<table>
<tr><th>Rank</th><th>Score</th><th align=left>Title/Information</th><tr>
<% foreach doc in result.documents %> <tr>
    <td valign=top align=center>$$doc.Rank</td>
    <td valign=top align=center>$$doc.Score</td> <td>
       <a href="$$doc.URL HTML"><strong>$$doc.Title</strong></a>
       <i><font size=-1><% Left(doc.snippet,240) %>...</font></i> </td> </tr>
<% endfor %>
</table>
<I>Template: $$Template</I>
</html>
```

ADVANCED RESULTS

This template adds user-definable results options

- Sorting
- Threshold
- Selection of other results templates



BASIC SERVER CONFIGURATION

Server Setup allows you to configure how resources are to be allocated to the search engine and the defaults to be used for searching and finding files. Performance tuning allows you to specify how searching will be done on the server. Simple settings allow you to select local, single department or entire enterprise and advanced settings let you enter the maximum number of searches per hour, simultaneous requests, and search sessions.

Setting search defaults allows you to indicate:

- The number of documents to be listed on each page
- How sorting should be handled
- What threshold should be set to eliminate less relevant documents
- How highlighting and navigation are to be handled

Setting path defaults lets you ensure the common files will be easily found whenever the server needs them. When you enter settings through the Information Server, they are captured in the inetsrch.ini file. This is the file that controls your entire application.

Sample inetsrch.ini file (in your SEARCH'97\bin directory)

```
[Common]
Topicset=c:\is97\topics\news
topichome=c:\search97\common
ProductLongName=SEARCH'97 Information Server V3.1
ProductShortName=Information Server
PatchLevel=10
ProductHomeLink=http://www.verity.com
VTopicScriptName=/search97cgi/s97_cgi.exe
VTopicSScriptName=/search97securecgi/s97_cgi.exe
ProgName=c:\search97\_nti31\bin\s97_cgi.exe
charmap=8859
locale=english
VerityCopyright=1997
Authentication=turned-on
AdminUser=student
installdir=c:\search97\
InstallDate=4qR08PGiyJX6M
SrvrRoot=C:\Netscape\SuiteSpot\https-train2\config
SrvrType=Netscape Enterprise &Server v3.x
OperatingSystem=NT
OperatingSystemVersion=4.0
UseJavaScript=0
Slash=\
AgentDir=c:\search97\s97as
Tasdir=/search97/agents
BodyElements=
PcnConnect=None

[Common\ErrorDefaults]
ErrorImagePath=/search97admimg/
ErrorTemplate =

[Common\ActionList]

[Common\ActionList\Actions(1)]
```

```
name = ErrorReport
spec = vformat
configSection = ErrorDefaults

[S97IS\ServerMap]
*:* = Primary

[S97IS\ServerMap\Primary]

[S97IS\Primary\Server]
Search97ImagePath=/search97img/
Sessions=3
BeginHighlight=<strong>
EndHighlight=</strong>
QueryMode=Simple
ResultCount=10
ScoreThreshold=0
Proxy=
ProxyPort=
BodyElements=bgcolor=ffffff
TemplateDir(1)=c:\is97\samples\template
TemplateDir(2)=c:\search97\s97is\locale\english\samples\template\
TemplateDir(3)=c:\search97\s97is\locale\english\agents\template
TemplateDir(4)=c:\search97\s97is\locale\english\agents\template\user
TemplateDir(5)=c:\search97\s97is\locale\english\agents\template\user2
DefSearchTemplate=pwersrcp.hts
Company=Pauls Company

[S97IS\Primary\SearchDefaults]
ResultTemplate = default.hts
ResultErrorTemplate = serror.hts
Filter = filter.hts

[S97IS\Primary\ViewDefaults]
ViewErrorTemplate = serror.hts
DynamicHLLevel=1
;DynamicHLLevel can be 0 (use collection info and don't scan the document
;to find terms for highlighting), 1 (fast scan), and 2 (accurate but slow)
=;to find terms for highlighting), 1 (fast scan), and 2 (accurate but slow)

[S97IS\Primary\FormGenDefaults]
Template = smplsrcp.hts
;;[S97IS\Primary\RedirDefaults]
;;redirPort = 8990
;;redirScript = /cgi-bin/vtopic
;;redirServiceName = Search97 Information Server
;;redirServiceProg = c:\search97\_nti31\admin\vhttp.exe
;;redirServiceTmpDir = c:\search97\s97is\locale\english\system
;;redirServiceArgs(1) = -port
;;redirServiceArgs(2) = 8990
;;redirServiceArgs(3) = -workers
;;redirServiceArgs(4) = 6
;;redirServiceArgs(5) = -vtopicConfig
;;redirServiceArgs(6) = -configPath
;;redirServiceArgs(7) = c:\search97\_nti31\bin\inetsrch.ini
=;;redirServiceArgs(7) = c:\search97\_nti31\bin\inetsrch.ini

[S97IS\Primary\ActionList]

[S97IS\Primary\ActionList\Actions(1)]
name = Search
spec = vsearch,vformat
configSection = SearchDefaults
```

```
[S97IS\Primary\ActionList\Actions(2)]
name = View
spec = vdocview,vformat
configSection = ViewDefaults

[S97IS\Primary\ActionList\Actions(3)]
name = FormGen
spec = vformat
configSection = FormGenDefaults

[S97IS\Primary\ActionList\Actions(4)]
name = FilterSearch
spec = vfilter,vsearch,vformat
configSection = SearchDefaults

[S97IS\Primary\FileMap(1)]
FileMapDir=c:\search97\s97is\locale\english\
ServerMapDir=/search97/

[S97IS\Primary\CollectionList]

[S97IS\Primary\CollectionList\Collections(1)]
Alias=C1
Description=SEARCH'97 Information Server Documentation
Name=Documentation Collection
Path=c:\search97\s97is\colls\help.clm
State=Enabled
ReadOnly=True
DefaultList=True
Profile=True

[S97IS\Admin]
;;SearchServiceType=Server
;;SearchServerPortal=/s97is.vts
AdminServiceType=Server
ShowWelcome=True
TemplateDir(1)=c:\search97\s97is\locale\english\admin\
TemplateDir(2)=c:\search97\s97is\locale\english\samples\template\
UseFrames=True
AdminImagePath=/search97admimg/
AdminScriptname=/s97a.vta
AdminSScriptName=/search97secure/s97a.vta
AdminProdNotes=/search97/doc/prodnote/
AdminUserGuide=/search97/doc/user/
AdminSearchScriptGuide=/search97/doc/srchscr/
AdminCollectionBuilding=/search97/doc/collbldg/
SearchTips=/search97/doc/tips/
DefCollPath=c:\search97\s97is\colls
Filter=adminsec.hts
=;;SearchServerPortal=/s97is.vts
Tasdir=/search97/agents
TemplateDir(3)=c:\search97\s97is\locale\english\agents\template
TemplateDir(4)=c:\search97\s97is\locale\english\agents\template\admin
TemplateDir(5)=c:\search97\s97is\locale\english\agents\template\user
TemplateDir(6)=c:\search97\s97is\locale\english\agents\template\user2

[S97IS\Admin\RedirDefaults]
redirPort = 8989
redirScript = /cgi-bin/vtopic
redirServiceName = Search97 IS Admin
redirServiceProg = c:\search97\_nti31\admin\vhttp.exe
redirServiceTmpDir = c:\search97\s97is\locale\english\system
```

```
redirServiceArgs(1) = -port
redirServiceArgs(2) = 8989
redirServiceArgs(3) = -configPath
redirServiceArgs(4) = c:\search97\_nti31\bin\inetsrch.ini

[S97IS\Admin\ActionList]

[S97IS\Admin\ActionList\Actions(1)]
ConfigKey=
ConfigPath=
ConfigSection=
Name=Admin
Spec=vfilter,Admin,Formatter

[S97IS\Admin\ActionList\Actions(2)]
ConfigKey=
ConfigPath=
ConfigSection=
Name=ServerManager
Spec=vfilter,ServerManager,Formatter

[S97IS\Admin\ActionList\Actions(3)]
ConfigKey=
ConfigPath=
ConfigSection=
Name=IndexManager
Spec=vfilter,IndexManager,Formatter

[S97IS\Admin\ActionList\Actions(4)]
ConfigKey=
ConfigPath=
ConfigSection=
Name=CollManager
Spec=vfilter,CollManager,Formatter

[S97IS\Admin\ActionList\Actions(5)]
ConfigKey=
ConfigPath=
ConfigSection=
Name=FormWizard
Spec=vfilter,FormWizard,Formatter

[S97IS\Admin\ActionList\Actions(6)]
ConfigKey=
ConfigPath=
ConfigSection=
Name=Formatter
Spec=vfilter,Formatter

[S97IS\Admin\ActionList\Actions(7)]
ConfigKey=
ConfigPath=
ConfigSection=
Name=AllAdmin
Spec=vfilter,Admin,ServerManager,IndexManager,CollManager,FormWizard,Formatter

[S97IS\Admin\ComponentList]

[S97IS\Admin\ComponentList\Components(1)]
Name=Admin
Spec=LINK:0:AdminService

[S97IS\Admin\ComponentList\Components(2)]
```

```
Name=ServerManager
Spec=LINK:0:SrvrMngrService

[S97IS\Admin\ComponentList\Components(3)]
Name=IndexManager
Spec=LINK:0:IndxMngrService

[S97IS\Admin\ComponentList\Components(4)]
Name=CollManager
Spec=LINK:0:CollMngrService

[S97IS\Admin\ComponentList\Components(5)]
Name=FormWizard
Spec=LINK:0:FormWzrdService

[S97IS\Admin\ComponentList\Components(6)]
Name=Formatter
Spec=LINK:0:FormatterService

[S97IS\Admin\ComponentList\Components(7)]
Name=vfilter
Spec=LINK:0:vfilter

[S97IS\FormsWizard]

[S97IS\FormsWizard\Themes(1)]
Name=Simple
SearchTemplate=smplsrcp.hts

[S97IS\FormsWizard\Themes(2)]
Name=Standard
SearchTemplate=stndsrcp.hts

[S97IS\FormsWizard\Themes(3)]
Name=Power
SearchTemplate=pwersrcp.hts

[S97IS\FormsWizard\Themes(4)]
Name=Advanced-Select
SearchTemplate=advsel.hts

[S97IS\FormsWizard\Themes(5)]
Name=Advanced-All-In-One
SearchTemplate=a1search.hts

[S97IS\Admin\Menu]

[S97IS\Admin\Menu\Items(1)]
Image=quikstrt.gif
Name=Quick Start
Query=Action=Formatter&Template=quikstrt.hts

[S97IS\Admin\Menu\Items(2)]
Image=srvradmn.gif
Name=Server Setup
Query=Action=ServerManager&Command=GetMenu

[S97IS\Admin\Menu\Items(3)]
Image=indxmngr.gif
Name=Indexing Manager
Query=Action=IndexManager&Command=GetMenu

[S97IS\Admin\Menu\Items(4)]
```

```
Image=collmngr.gif
Name=Collection Manager
Query=Action=CollManager&Command=GetMenu

[S97IS\Admin\Menu\Items(5)]
Image=formwzrd.gif
Name=Forms Wizard
Query=Action=FormWizard&Command=GetMenu

[S97IS\Admin\Menu\Items(6)]
Image=userguid.gif
Name=User Guide
Query=Action=Formatter&Template=userguid.hts

[S97IS\Admin\IndexManager]
sname=UniversalSpider
sspec=LOCALHOST
LastJob=c:\is97\colls\legal

[S97IS\Admin\IndexManager\UniversalSpider]
LogLevel=Verbose
MaxDocLen=0
;MaxDocLen controls the maximum size of document to index.
;Setting this to zero, means the max. size allowable is unlimited
=;Setting this to zero, means the max. size allowable is unlimited

[Common\DeliveryDefaults]
DlvConnCount=2
Filter=tasflt.hts
DlvMaxDocsPerAgent=3
DlvSort=agentid asc
DlvVerb=Search

[Common\AgentDefaults]
AgtConnCount=2
Filter=tasflt.hts
AgtSort=name asc
SyncAgentCount=0
AgtSearchRange=10
AgtSearchBase=0

[Common\UserDBDefaults]
USERDBSCHEMA=firstname,lastname,maxagents,curagents,usercollection/StringArray,userdel
ivery/StringArray,useremail,pageremail,userwebform,expiration,disabled,passwd,accessqr
y
USEAUTHDB=True
AUTHDBSERVER=Netscape
AUTHSRVRROOT=C:\Netscape\SuiteSpot
STORECLEARTEXTPASSWORDS=False
AUTHDBENCRYPT=DES
AUTHDBTYPE=pwfile
AUTHDBPATH=c:\search97\userdb\verity.pwf
Filter=tasflt.hts
udbverb=list
USERDBPATH=c:\search97\userdb\agentudb
UdbSearchRange=10
UdbSearchBase=0

[Common\FilterFormatDefaults]
Filter=tasflt.hts

[Common\ComponentList\Components(1)]
name=vuserdb
```

```
spec=DLL:c:\search97\_nti31\bin\libudb:UserDBService

[Common\ComponentList\Components(2)]
name=vagent
spec=DLL:c:\search97\_nti31\bin\libvtas:AgentService

[Common\ComponentList\Components(3)]
name=vdelivery
spec=DLL:c:\search97\_nti31\bin\libvtas:DeliveryService

[Common\ComponentList\Components(4)]
name=vpcn
spec=DLL:c:\search97\_nti31\bin\vpcn:PcnService

[S97IS\Primary\ActionList\Actions(5)]
name=FilterFormat
configSection=FilterFormatDefaults
spec=vfilter,vuserdb,vformat

[S97IS\Primary\ActionList\Actions(6)]
name=userdb
configSection=UserDBDefaults
spec=vfilter,vuserdb,vformat

[S97IS\Primary\ActionList\Actions(7)]
name=agent
configSection=AgentDefaults
spec=vfilter,vuserdb,vagent,vformat

[S97IS\Primary\ActionList\Actions(8)]
name=delivery
configSection=DeliveryDefaults
spec=vfilter,vuserdb,vdelivery,vformat

[S97IS\Primary\ActionList\Actions(9)]
name=FilterSearchFormat
configSection=FilterFormatDefaults
spec=vfilter,vsearch,vformat

[S97IS\Primary\TIA\DlvActions\DlvAction(1)]
Name=Web
State=Enabled
ActionArgs=html AGENT_NAME=${AGENT_NAME} TITLE=${TITLE} SUBJECT=${SUBJECT}
AGENT_QUERYQUESTION=${AGENT_QUERYQUESTION}
AGENT_QUERYTHRESHOLD=${AGENT_QUERYTHRESHOLD} MIME-TYPE=${MIME-TYPE}
DeliveryMethod=database
Description=My News Page

[S97IS\Primary\TIA\DlvActions\DlvAction(1)\Input(1)]
type=none
name=Userwebform

[S97IS\Primary\TIA\DlvActions\DlvAction(1)\Input(1)\Value(1)]
name=Frames
description=Frames Display

[S97IS\Primary\TIA\DlvActions\DlvAction(1)\Input(1)\Value(2)]
name=No Frames
description=No Frames Display

[S97IS\Primary\TIA\DlvActions\DlvAction(2)]
Name=Email
State=Enabled
```

```
ActionArgs=To=$$useremail,Subject=${AGENT_NAME}: ${TITLE},Body=${_$}
DeliveryMethod=smtp
Description=Electronic Mail

[S97IS\Primary\TIA\DlvActions\DlvAction(2)\Input(1)]
type=textedit
value=
name=useremail

[S97IS\Primary\TIA\DlvActions\DlvAction(3)]
Name=Pager
State=Enabled
ActionArgs=To=$$pageremail,Subject=${AGENT_NAME}: ${TITLE},Body=${TITLE}
DeliveryMethod=smtp
Description=Pager

[S97IS\Primary\TIA\DlvActions\DlvAction(3)\Input(1)]
type=textedit
value=
name=pageremail

[S97IS\Primary\TIA\DLVConfigList\DLVConfigArray(1)]
DlvDriverName=Verity Repository Driver
DlvPerfButton=3
RepositoryBased=True
DlvHitDays=7
DlvKeyArray(1)=dda
DlvKeyArray(2)=repository
DlvKeyArray(3)=style
DlvKeyArray(4)=background
DlvKeyArray(5)=mode

[S97IS\Primary\TIA\DLVConfigList\DLVConfigArray(1)\Values]
dda=DLL:c:\search97\_nti31\bin\libdvdk:DeliveryMain
mode=NewsFeedIdx
background=off
style=c:\search97\s97is\locale\english\styles\dlvstyle
repository=c:\search97\s97as\v_store

[S97IS\Primary\TIA\DLVConfigList\DLVConfigArray(2)]
DlvDriverName=Email Driver (SMTP)
DlvPerfButton=3
DlvKeyArray(1)=dda
DlvKeyArray(2)=smtp_hostname
DlvKeyArray(3)=from
DlvKeyArray(4)=default_to
DlvKeyArray(5)=max_mail_size

[S97IS\Primary\TIA\DLVConfigList\DLVConfigArray(2)\Values]
dda=DLL:c:\search97\_nti31\bin\libdsmtp:DeliveryMain
max_mail_size=64000
default_to=AgentServer
from=AgentServer
smtp_hostname=localhost

[S97IS\Primary\TIA\DLVConfigList\DLVConfigArray(3)]
DlvDriverName=Email Driver (MAPI)
DlvPerfButton=3
DlvKeyArray(1)=dda
DlvKeyArray(2)=user
DlvKeyArray(3)=password
DlvKeyArray(4)=default_to
DlvKeyArray(5)=login_ui
```

```
[S97IS\Admin\TIA]
TIAConfigPath=c:\search97\_nti31\bin\inetsrch.ini

[S97IS\Admin\TIA\Menu\Items(1)]
Image=/search97/agents/images/awelcome.gif
Query=Action=FilterFormat&Template=whatsnew.hts
Name=Whats New

[S97IS\Admin\TIA\Menu\Items(2)]
Image=/search97admimg/locale/english/admin/srvradmn.gif
Query=Action=FilterFormat&Template=srvrmain.hts
Name=Server Setup

[S97IS\Admin\TIA\Menu\Items(3)]
Image=/search97admimg/locale/english/admin/collmngr.gif
Query=Action=CollManager&Command=GetMenu
Name=Collection Manager

[S97IS\Admin\TIA\Menu\Items(4)]
Image=/search97/agents/images/usermngr.gif
Query=Action=userdb&udbverb=list&Template=usermngr.hts&UdbSearchRange=-1
Name=User Manager

[S97IS\Admin\TIA\Menu\Items(5)]
Image=/search97admimg/locale/english/admin/userguid.gif
Query=Action=FilterFormat&Template=userguid.hts
Name=User Guide

[S97IS\Admin\Primary\TIA]
TasSvcType=Service
ProfPerfButton=4

[S97IS\Admin\Primary\TIA\vdkwatch]
ConfigPath=c:\search97\s97as\vdkwatch.cfg
PostAgents=
XORAgents=
RegularAgents=
RequireColls=False
PIDFile=c:\search97\s97as\logs\vdkwatch.pid
StateFile=c:\search97\s97as\vdkwatch.sta
GlobalStyle=c:\search97\s97is\locale\english\styles\unistyle
LicenseKey=not applicable for v2+ kernel
Batch=0
PollInterval=300
ServiceInterval=120
OutLevel=1
LogFile=c:\search97\s97as\logs\vdkwatch.log
LogLevel=15
ignoreBadColls=NO

[S97IS\Admin\Primary]

[S97IS\Primary]

[S97IS\Primary\CollectionList\Collections(2)]
Alias=verity
Name=verity
Path=c:\search97\s97is\colls\verity.clm
Description=Verity Web Site
State=Enabled
DefaultList=True
ReadOnly=False
```

```
Profile=False

[S97IS\Primary\CollectionList\Collections(3)]
Alias=legal
Name=legal
Path=c:\is97\colls\legal.clm
Description=practice lab #2
State=Enabled
DefaultList=True
ReadOnly=False
Profile=False

[S97IS\Primary\CollectionList\Collections(4)]
Alias=misc
Name=misc
Path=c:\is97\colls\misc.clm
Description=news topics
State=Enabled
DefaultList=True
ReadOnly=False
Profile=True
```

PRACTICE LAB #1A
REVIEWING THE INFORMATION SERVER INSTALLATION

In this lab you will review what the environment looks like, after the Information Server has been installed.

1. Prior to set up, an http server instance was created for your Information Server. Take a few minutes to explore this instance. For center classes we use a Netscape Enterprise Server. Your instructor will provide server information for onsite courses. Launch your web browser and enter the appropriate URL:

   http://train1:8001 (Substitute the 1 after train with your own student id number)
   username: student
   password: learn

   Click on the `train1` button to access your web server. Then click on the "View Server Settings" from the Server Preferences list. The technical settings (magnus.conf), shows that the port being used for your "train1" server is **7000.** Scroll down and notice under content settings (obj.conf), that there are many additional references to where key information about the Information Server can be found. Here you will see that the web server knows about the admin directory (which has all the Information Server administration components), the images directory which has default gifs used in building server pages, the search97 (s97is) directory which is pathed to your install area, the cgiadm and bin directories with executables, and that we have set up a samples directory where you can use any of the training template files for searching and viewing documents. The /samples directory is mapped to c:/is97/samples/template which will hold all of your new search and results forms.

2. Review the installation files (on the file system) and the special training environment. In Center classes you will find the Verity software at c:\search97 and the training environment at c:\is97. If this course is being taught onsite, please ask your instructor for the correct path to the install directory and to the training samples. Write them here:

   Installation Directory: _____

   Training Samples Directory: _____

3. Review the configuration file: inetsrch.ini, which you will find in the under the install area, under the bin directory. For example: c:\search97_nti31\bin.

   Some important points to review in this file include:
   - Common and primary settings
   - Where template directories are specified
   - Collections that have been added to the server

Practice Lab #1B
Using the Quick Start

4. Now start the Verity Information Server by entering this URL (of course substituting your student information for train1):

   http://train1:7000/search97

   This page provides access to the Admin Server (which you will log into in just a moment) and to the documentation resources available to you. It also has some sample search forms that might be of use to you. Take a few minutes to check out the documentation that is provided.

   Click on the URL to administer your server. You can either use this link or enter the URL directly: http://train1:7000/search97secure/s97a.vta**?**

5. When you have the Information Server's welcome page, please create a bookmark to this page. You will return many times.

6. Click the Configure Server button to move you to the Quick Start. The Quick Start consists of three parts. The first one allows you to check your server to make sure it is properly configured.

7. Check your system configuration (this may take a moment) and make sure you received a go-ahead that no errors were found. If errors were found, please notify your instructor.

8. Now you can index our web site. Enter http://www.verity.com in the "URL to Index" field, then click the "New" button to create an empty collection directory. Remember that the collection needs to be created first, so there will be a "container" to store the information that is captured during the indexing process.

9. Enter "verity" as the name of your new collection and enter "Verity's Web Site" in the Description field. Read through the information shown above the "Path" box. As you see, there is a default area where collections are created. Because we used "verity" as the name of the collection, which meets the guidelines of being up to 8 characters without spaces, the collection will be found on your system, using the default path and will be called verity. If you want collections to go elsewhere on your system, you can simply enter the full path to the area where you wish the collection to reside. For example: c:\is97\colls\verity

   Click the "Create" button and in a moment you will see the following message: "The collection verity has been added (but is currently empty)." You can now click the OK button to move back to the "Index Your Web Site" form. You should now see your collection name appears in the Destination field.

10. You are now ready to actually begin the indexing. Click the "Index" button and the Information Server will display the "Site Indexing" form. Notice that it shows a summary of the indexing status for the site you are indexing. This status indicates whether the indexer is currently working or has finished, and it displays the number of documents indexed. You may click the "Update" button to update the status as it is processing. If the status indicates a problem, use the "View Log" icon to see a detailed listing of the indexing activity. Read through any error messages and if you're not sure what has gone wrong, please inform your instructor about the problem.

11. When indexing has finished, go back to the Quick Start page and click on Step 3, "Search Your Web Site" to display a simple search form. You can choose your collection by clicking on the collection pop-down list and selecting it. Notice that you also have the Verity Documentation Collection here. This is the default that ships with the product.

12. Enter the word "solution" in the query field and press Enter or click on the Search button. The Results list will display the documents that match your query. View some of the documents and notice the word is highlighted in the documents. Practice with a few queries of your own, but remember you can only find results on matching words so take a look at a few of the documents to see what these collections are about. A little later today you will learn about the full query language but for now, try a few using the basic operators: AND and OR. Choose your own words from the documents you have selected: Enter your query using any one of these combinations: **word1 AND word2**, **word1 and word2**, **word1 <and> word2**. Now substitute the AND for OR.

13. Practice with searches against the individual collections as well as all of the collections. Consider what this list will look like at your site. Jot down ideas about how you will probably organize your information (in collections) and be prepared to share them with the class.

INDEXING WITH THE VERITY SPIDER

The Information Server's indexing spider can create collections from a variety of document types, including:

- ASCII text documents. This includes text documents and documents in Internet e-mail or unused news format.
- HTML documents
- Adobe Acrobat documents. This includes documents in Acrobat PDF format.
- WSIWYG documents. This includes documents supported by the MasterSoft filter and viewer kit, including Microsoft word, WordPerfect and other formats.

The spider can automatically index all of these various document types into a single master collection. While internally each document type is indexed into an appropriate collection for it's type, the end result appears to be a single collection. This greatly simplifies collection management and making collections available to users.

WHAT IS A COLLECTION?

Collections represent groups of documents that have been processed so they are retrievable by Verity Products. During processing, optimized indexes are automatically created to enable fast searching of words contained in the documents or attribute information about the documents like their titles or authors. A collection is a series of indexes, which store data about your documents.

- Some HTML tags are translated automatically into zones for more specific searching
- Zones allow full query language searching (including topics)
- Fields can be also be created for more targeted searching or for use in sorting or displaying results

All the words in the document are captured and include information to allow concept, Boolean and proximity searching.

THE COLLECTION DIRECTORY

- Each time a new collection is created, a home directory is created
- The collection home directory includes sub-directories that store and maintain information about the collection, style files and documents

The collection directory contains all the files that are needed to provide access to your documents or to manage the collection. The main job of the collection is to store information about the document (attributes) and to store the word index. Because you index at different times, or index a large amount of data, the collection is optimized into smaller components called partitions, each having these two indexes (attributes and words) for a group of documents in the

collection. A collection can have many partitions. Depending upon how the collection was created, there are other directories in the collection that store data. Here is an overview of the basic role of these various collection directories. Remember that all collections do not have all of the same sub-directories, they only have the ones they need.

The **ASSISTS Directory** stores spanning word lists and information about the collection. The **PARTS Directory** stores partitions, consisting of two key elements, the ".ddd" which holds the attribute information and the ".did" which holds the word index.

The .ddd file is a binary list of all of the documents in a partition. There is a separate record for each document, which includes several internal fields (like the document name) along with user-defined fields. The .did file is a binary database of all the words in all of the documents in a partition. If the documents are large and the partition is large, the .did files will be quite large as well. The general rule of thumb is that the .did will probably be about 40% of the size of the original documents.

The **PDD Directory** stores the partition index, which manages all the partitions in the collection. The **STYLE Directory** stores configuration files. The **TRANS Directory** stores transaction files used in auto-administration.

Other collection directories you may find, based on the collection utility's needs or features, are the **TOPICIDX Directory** which stores topic indexes for each partition, **WORK Directory** which stores task files used in auto-administration, **MORGUE Directory** which stores files that will be deleted.

THE ROLE OF STYLE FILES

- Style files configure the series of indexes storing data about documents and your environment
- The choices made in style files direct how indexing utilities will create indexes and the guidelines for doing work
- The Information Server ships with a special style directory which allows you to incorporate many types of documents in your collection, without having to do any additional work

EXPLORING STYLE GUIDELINES

You will probably not need to alter your style files. Verity tries to anticipate the kinds of information you will wish to extract from documents and do the prep work for you. However, it is possible that your documents will have special information, which could not be anticipated. When this is the case, you will need to edit style files to add the information you want to extract. This class does not attempt to cover everything related to style files for all utility programs, and there is additional training available. If you need to make changes to style files, we suggest that you review the user guide called Collection Building. It contains the details you need to modify style files or to choose which style files you should incorporate based on your specific needs.

Just to give you an idea about how the style files are defined, here are a few that are used in the main style directory provided by Verity.

THE STYLE.DDD FILE CONFIGURES THE COLLECTION

```
# $Id: style.ddd,v 1.1.1.5 1997/02/23 04:19:02 wade Exp $
# Copyright (C) 1987-1997 Verity, Inc.
# Document Dataset Descriptor
$control: 1
$include style.prm
$subst: 1
descriptor:
  /collection = yes
{
  data-table:     _df
    /num-records = 1
    /max-records = 1
  {
    # Header information for partition management
    worm:   _DBVERSION  text
    fixwidth:     _DDDSTAMP   4 date
    varwidth:     _DOCIDX           _dv
    fixwidth:     _PARTDESC   32 text
    constant:   _FtrCfg         text "${DOC-FEATURES:}"
    constant:   _SumCfg         text "${DOC-SUMMARIES:}"

    fixwidth:   _SPARE1         16 text
    fixwidth:   _SPARE2         4 signed-integer
  }
  # Required internal fields per document
  data-table:     _df
    /offset=64
  {
    autoval:      _STYLE            sirepath
    fixwidth:     _DOCID            4 unsigned-integer
    fixwidth:     _SECURITY   4 unsigned-integer
      /minmax = yes
    fixwidth:     _INDEX_DATE 4 date
      /minmax = yes
  }
$ifdef DOC-FEATURES
  # Optional feature vector per document
  data-table:   _dg
  {
    varwidth:   VDKFEATURES     _dh
      /_implied_size
  }
$endif
$ifdef DOC-SUMMARIES
  # Optional generated summary per document
  data-table:   _di
  {
    varwidth:   VDKSUMMARY      _dj
```

```
      /_implied_size
  }
$endif
$include style.ufl
}
$$
```

THE STYLE.DFT FILE DEFINES THE VIRTUAL DOCUMENT

HTML Version

```
# $Id: style.dft,v 1.1.1.5 1997/04/17 00:04:43 ameyer Exp $
# Copyright (C) 1987-1995 Verity, Inc.
# Document Format
#
$control:1
dft:
{
  field: DOC
    # The following line invokes the HTML zone filter
    /filter="zone -html"
}
$$
```

Auto-Filt Version

```
# $Id: style.dft,v 1.1.1.1 1996/09/12 22:46:56 ameyer Exp $
# Copyright (C) 1987-1995 Verity, Inc.
# Document Format
#
$control:1
dft:
{
  field: DOC
    # The following line invokes MasterSoft WYSIWYG filtering
    /filter = "auto"
}
$$
```

PDF Version

```
# $Id: style.dft,v 1.1.1.3 1997/02/28 19:14:57 wade Exp $
# Copyright (C) 1987-1996 Verity, Inc.
#
# Document Format
control:1
dft:
{
  field: DOC
    /filter="flt_pdf -charmapto 850"
}
$$
```

THE STYLE.PRM FILE GOVERNS WORD INDEXING FEATURES

```
# $Id: style.prm,v 1.1 1997/02/23 04:26:15 wade Exp $
# Copyright (C) 1987-1995 Verity, Inc.
#
# style.prm - collection schema parameters
#
# This file is used to enable/disable index schema features through
# macro definitions similar to those allowed by the C preprocesser.
# This file is included in other style files using $include so
# that the selected features are propogated to the schemas of all
# tables in the index.  Refer to the "Using the style.prm File"
# chapter in the Collection Buiding Guide for more information.

# ------------------------------------------------------------------
# The IDX-CONFIG parameter defines the storage format used to
# encode the word positions in the index.  WCT (Word Count) format
# is a compact format, storing the ordinal counting position of the
# word from the beginning of the document.  PSW (Paragraph, Sentence,
# Word) format takes approximately 15-20% more disk space, but
# stores semantically accurate paragraph and sentence boundaries.
# Optionally, Many may be specified with either WCT or PSW to
# improve the accuracy of the <MANY> operator at the expense of
# diskspace and search performance.

# This example enbles Word Count word position format (the default).
$define IDX-CONFIG   "WCT"

# This example turns on Paragraph/Sentence/Word word position format.
# It also enables the <MANY> operator accuracy improvement.
#$define IDX-CONFIG "PSW Many"

# ------------------------------------------------------------------
# The IDXOPTS parameters define which index options are applied to
# the various index token tables.  The following index options are
# supported for each: Stemdex enables an index by the stem of each
# word. Casedex stores all case variants of a word separately, so
# one can search for case sensitive terms such as "Jobs", "Apple",
# and "NeXT" more easily. Soundex stores phonetic representations
# of the word, using AT&T's standard soundex algorithm.  The
# application may also store 1-4 bytes of application-specific
# data with each word instance, in the form of Location data and/or
# Qualify Instance data.  These options are specified separately
# for each token table: word, zone, and zone attribute.
$define WORD-IDXOPTS       "Stemdex Casedex"
$define ZONE-IDXOPTS       ""
$define ATTR-IDXOPTS       "Casedex"

# ------------------------------------------------------------------
# Clustering is enabled by uncommenting the DOC-FEATURES line.
# This stores a feature vector for each document in the
# Documents table.  These features are used for Clustering
# results and fast Query-by-Example.  See the discussions on
# Clustering in the Collection Building Guide for more information.
$define DOC-FEATURES    "TF"
```

```
# ------------------------------------------------------------------
# Document Summarization is enabled by uncommenting one of
# the DOC-SUMMARIES lines below.  The summarization data is
# stored in the documents table so that it might easily be
# shown when displaying the results of a search.
# See the discussions on Document Summarization in the
# Collection Building Guide for more information.
# The example below stores the best three sentences of
# the document, but not more than 500 bytes.
$define DOC-SUMMARIES   "XS MaxSents 3 MaxBytes 500"
# The example below stores the first four sentences of
# the document, but not more than 500 bytes.
#$define DOC-SUMMARIES   "LS MaxSents 4 MaxBytes 500"
# The example below stores the first 150 bytes of
# the document, with whitespace compressed.
#$define DOC-SUMMARIES   "LB MaxBytes 150"
```

The Style.WLD Governs How Words Are Stored

```
# $Id: style.wld,v 1.1.1.3 1997/04/02 23:26:47 wade Exp $
# Copyright (C) 1987-1997 Verity, Inc.
# style.wld - Word List Descriptor
$control: 1
$include style.prm
$subst: 1
descriptor:
{
  data-table:    _ph
    /num-records=1
    /max-records=1
  {
    # Header information for partition management
    worm:         _DBVERSION     text
    fixwidth:     _DDDSTAMP      4  date
    fixwidth:     _DIDSTAMP      4  date
    constant:   Types           text "Word Zone Attr"
    constant:   Config          text "$IDX-CONFIG"
    constant:   Word            text "$WORD-IDXOPTS"
    constant:   Zone            text "$ZONE-IDXOPTS"
    constant:   Attr            text "$ATTR-IDXOPTS"
    constant:   TSPARE1          text ""
}
# This is the table of parts covered by spanning wordlist
  data-table:    _pp
  {
    fixwidth:    PARTNUM     4 unsigned-integer
  }
  # The Actual full text index data
  data-table:   _pf
  {
    varwidth:   FWTEXT          _pw
      /_implied_size
    varwidth:   FWDATA          _pv
      /_implied_size
    fixwidth:   FWENCODE         1 unsigned-integer
    fixwidth:   FWFREQ          2 unsigned-integer
  }
```

```
  # The Btree for fast word lookup
  DATA-TABLE:   _PB
  {
    FIXWIDTH:   FWBTREE         3 TEXT
  }
# The accelerator for lookup by Stem
  DATA-TABLE:   _PS
  {
    FIXWIDTH:   STEMDEX         4 UNSIGNED-INTEGER
  }
# The accelerator for lookup by Soundex
  DATA-TABLE:   _PX
  {
    FIXWIDTH:   SOUNDEX         4 UNSIGNED-INTEGER
  }
}
$$
```

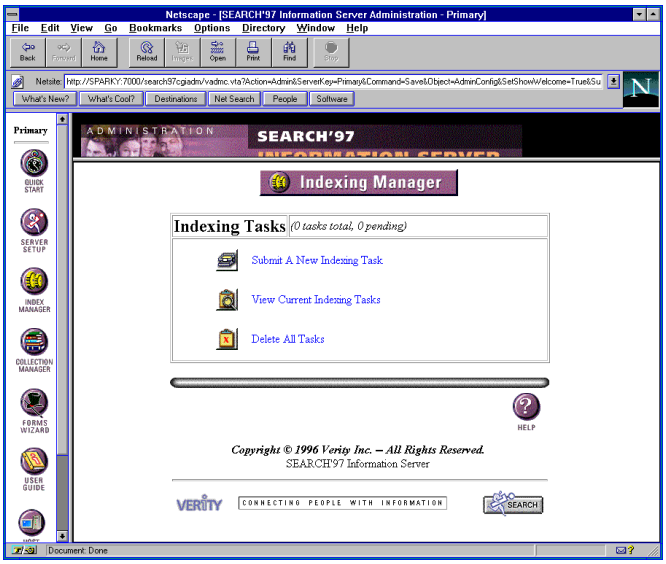
The Style.UFL File Governs User Defined Fields

```
# $Id: style.ufl,v 1.1 1997/02/23 04:26:39 wade Exp $
# style.ufl - Application-specific User Fields
#
  DATA-TABLE:    DKF
  {
    # Must have a VdkVgwKey - this is the primary key to the document
    # The field type can be changed to fixwidth integer or text as needed
    VARWIDTH:    VdkVgwKey   DKV
      /INDEXED = YES
      /MINMAX  = YES
  }
  DATA-TABLE:    DDF
  {
    # For file system collections:
    DISPATCH: DOC
    VARWIDTH: DOC_FN        DDG
    # For vspider
    VARWIDTH: _CACHE_FN     DDG
    FIXWIDTH: _CACHE_DELETE 1 UNSIGNED-INTEGER
    # For exchange
    VARWIDTH: _ParentID     DDH
    # Standard fields
    VARWIDTH: Title         DDH
    VARWIDTH: Ext           DDH
    VARWIDTH: Author        DDH
    VARWIDTH: Subject       DDH
    VARWIDTH: Keywords      DDH
    VARWIDTH: Comments      DDH
    VARWIDTH: Snippet       DDH
    VARWIDTH: URL           DDH
    VARWIDTH: MIME-Type     DDH
    VARWIDTH: Language      DDH
    VARWIDTH: Encoding      DDH
    VARWIDTH: _Created      DDH
    VARWIDTH: _Modified     DDH
    FIXWIDTH: Created       4 DATE
    FIXWIDTH: Modified      4 DATE
    FIXWIDTH: Size          4 UNSIGNED-INTEGER
  }
}
```

USING THE GRAPHICAL COLLECTION TOOLS

There are two key areas in the Information Server that are oriented to creating and managing collections.

The **Indexing Manager** helps you to index your site or other sites of interest



Collections can be created

- Simple mode requires a name, description and path
- Advanced mode allows you to filter for specific document MIME types, exclude or include based on specific filename pattern matching, and set proxy information at the collection level

Indexing tasks are defined and can be submitted for initial creation and then maintained as a current task for resubmission

The **Collection Manager** allows you to

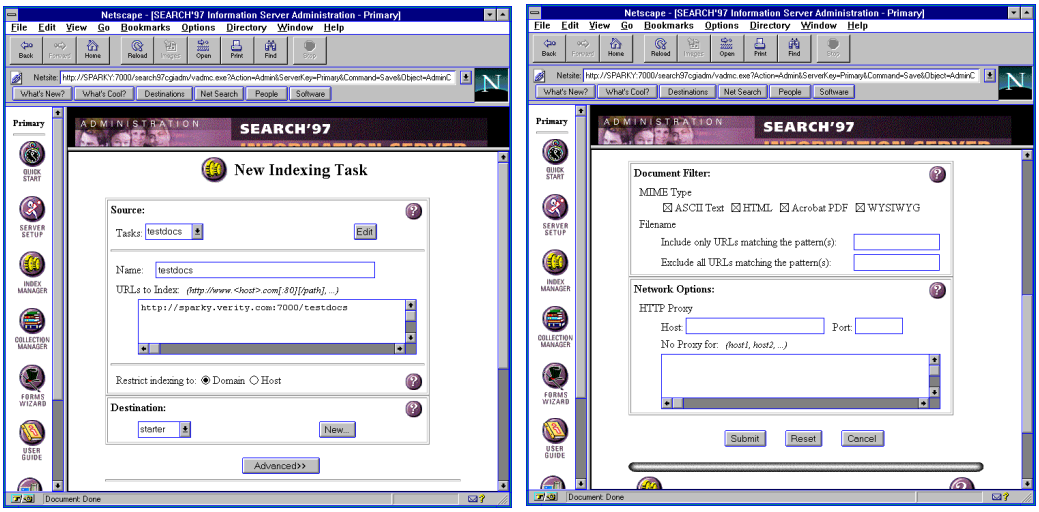
- Enable or disable collections for searching
- Import externally created collections
- View information about your collections

SUBMITTING INDEXING TASKS

- Before you can submit URLs for indexing, you must have a collection ready to receive the information
- Once you have a collection ready to go, you submit an indexing task which includes a name for the task, a list of URLs to index and the destination collection
- As you build your list of tasks, you will be able to select them from a drop-down list
- Be sure you index into the right collection because mistakes can be costly!

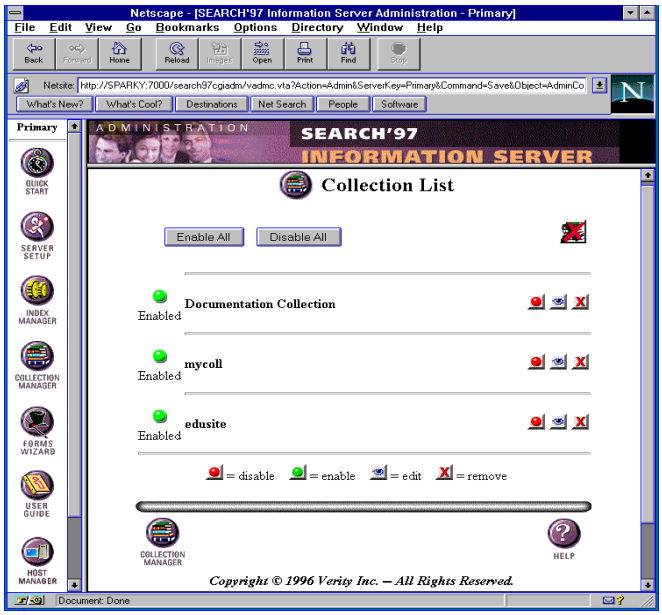
You can use the Indexing Manager to submit a new indexing task. To submit a new indexing task do the following:

1. Click the Indexing Manager button on the menu bar.
2. Click the New Indexing Task icon or hyperlinks on the Indexing Manager



USING THE COLLECTION LIST

As you build a number of collections, you will work with them on the Collection List. It is easy to enable and disable searching. The "Edit" icon takes you to Collection Properties.



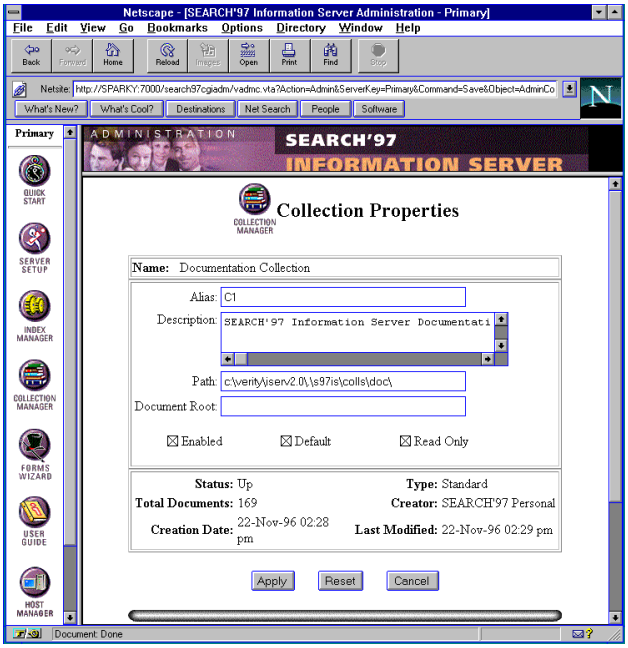
IMPORTING COLLECTIONS

If you have been using Verity products, you will have other collections that you wish to enable through The Information Server.

- Collections that have been built with a file indexer will need to have the Document Root field completed
- Collections built with previous versions of Verity's web indexers should import without a problem

REVIEWING COLLECTION PROPERTIES

- Properties provide information about the current status of the collection.



MORE ABOUT COLLECTIONS

The SEARCH'97 Information Server ships with additional collection and topic building and reporting tools

- mkvdk for building collections stored on the file system
- mktopics for building topics to be used in the Information Server
- browse to see the field and zone contents of your collection
- didump to see the words in your index

It is a good idea to become familiar with the features of these utilities, which are covered in detail in the Collection Building reference manual. This workbook contains a bit of information and exercises on each.

VIEWING DOCUMENT INFORMATION IN COLLECTIONS

There are two utilities you can use to better understand the information contained in your collection.

- Browse allows you to view values for each of the attributes captured in the .ddd file
- Didump allows you to view words in the word index (the .did) file

Here is information on using the Browse utility (found in the bin directory). You simply go to the partition directory (parts) under your collection, and choose which partition you wish to view. As appropriate for your platform enter this command:

```
browse 00000001.ddd
```

A short menu will appear. At this point you can press Enter and the first record will appear. If you have scroll bars, you can scroll back to view the field values. If not, just anticipate the commands and redirect output to a log file.

To do this, enter: browse `00000001.ddd > browse.out`

Press Enter, the first enter displays the menu (you won't see anything on the screen) and enter a second time to capture values for the first record, Enter again if you want to see the next record) and then type quit. You will be back at the prompt and can now view the log file (browse.out) to see the values.

```
?) help
 q) quit
 c) Number of entries in field
 _) Toggle viewing fields beginning with '_'
 v) Toggle viewing selected fields
##) Display all fields in specified record number
Dispatch/Compound field options:
 n) No dispatch
 d) Dispatch
 s) Dispatch as stream
Action (? for help):
```

Your log file should look like this.

```
browse.exe - Verity, Inc. Version 2.2.0 (_nti31, May  2 1997)

BROWSE OPTIONS
  ?) help
  q) quit
  c) Number of entries in field
  _) Toggle viewing fields beginning with '_'
  v) Toggle viewing selected fields
 ##) Display all fields in specified record number
Dispatch/Compound field options:
  n) No dispatch
  d) Dispatch
  s) Dispatch as stream

Action (? for help): Record number: 0

0  _DDFLAG          FIX-unsg (  1) = 0x00
1  _DDVALUE         VAR-text (  0) =
2  _DDVALUE_OF      FIX-unsg (  4) = 0
3  _DDVALUE_SZ      FIX-unsg (  2) = 0
4  _DBVERSION       WRM-text (  6) = vdk21
5  _DDDSTAMP        FIX-date (  4) = 06-May-1997 06:15:55 pm
6  _DOCIDX          VAR-text ( 12) = _
7  _PARTDESC        FIX-text ( 32) = mkvdk (Verity, Inc. Version 2.2.
8  _FtrCfg          CON-text (  3) = TF
9  _SumCfg          CON-text ( 27) = XS MaxSents 3 MaxBytes 500
10 _SPARE1          FIX-text ( 16) =
11 _SPARE2          FIX-sign (  4) = 0
12 _DOCIDX_OF       FIX-unsg (  4) = 32
13 _DOCIDX_SZ       FIX-unsg (  2) = 12
14 _STYLE           AUT-text ( 19) = ../style/style.ddd
15 _DOCID           FIX-unsg (  4) = 1
16 _SECURITY        FIX-unsg (  4) = 0
17 _INDEX_DATE      FIX-date (  4) = 06-May-1997 06:15:56 pm
18 _SECURITY_MI     WRM-unsg (  4) = 0
19 _SECURITY_MX     WRM-unsg (  4) = 0
20 _INDEX_DATE_MI   WRM-date (  4) = 06-May-1997 06:15:56 pm
21 _INDEX_DATE_MX   WRM-date (  4) = 06-May-1997 06:16:23 pm
22 VDKFEATURES      VAR-text (229) = _
23 VDKFEATURES_OF   FIX-unsg (  4) = 32
24 VDKSUMMARY       VAR-text (240) = This chapter covers basic information
about SEARCH'97(TM) Information Server and the SEARCH'97 search technology
from Verity© that is integrated in the product.  The following subjects are
included: Introduction to Verity Search Technology
25 VDKSUMMARY_OF    FIX-unsg (  4) = 32
26 VdkVgwKey        VAR-text ( 37) = ../locale/english/doc/user/01_is.htm
27 VdkVgwKey_IX     FIX-unsg (  3) = 123
28 VdkVgwKey_MI     WRM-text ( 41) = ../locale/english/doc/collbldg/01_bc.htm
29 VdkVgwKey_MX     WRM-text ( 40) = ../locale/english/doc/user/preface2.htm
30 VdkVgwKey_OF     FIX-unsg (  4) = 32
31 VdkVgwKey_SZ     FIX-unsg (  2) = 37
```

```
32 DOC              DSP-text ( -1) =
..\..\..\locale\english\doc\user\01_is.htm
33 DOC_FN           VAR-text ( 37) = ../locale/english/doc/user/01_is.htm
34 _CACHE_FN        VAR-text (  0) =
35 _CACHE_DELETE    FIX-unsg (  1) = 0
36 _ParentID        VAR-text ( 27) = ../locale/english/doc/user
37 Title            VAR-text ( 13) = Introduction
38 Ext              VAR-text (  4) = htm
39 Author           VAR-text (  0) =
40 Subject          VAR-text (  0) =
41 Keywords         VAR-text (  0) =
42 Comments         VAR-text (  0) =
43 Snippet          VAR-text (399) = Introduction . . . . . 1 . Introduction .
This chapter covers basic information about SEARCH'97(TM) Information Server
and the SEARCH'97 search technology from Verity&#174; that is integrated in
the product. The following subjects are included: . Introduction to
Information Server . New SEARCH'97 Features . Introduction to Verity Search
Technology . Copyright &#169; 1997, Verity, Inc. All rig...
44 URL              VAR-text (  0) =
45 MIME-Type        VAR-text ( 10) = text/html
46 Language         VAR-text (  0) =
47 Encoding         VAR-text (  0) =
48 _Created         VAR-text ( 26) = Wed May  7 01:04:19 1997
49 _Modified        VAR-text ( 26) = Wed Apr 30 00:09:02 1997
50 Created          FIX-date (  4) = 07-May-1997 01:04:19 am
51 Modified         FIX-date (  4) = 30-Apr-1997 12:09:02 am
52 Size             FIX-unsg (  4) = 1068
53 DOC_OF           FIX-unsg (  4) = 0
54 DOC_SZ           FIX-unsg (  4) = 4294967295
55 DOC_FN_OF        FIX-unsg (  4) = 32
56 DOC_FN_SZ        FIX-unsg (  2) = 37
57 _CACHE_FN_OF     FIX-unsg (  4) = 0
58 _CACHE_FN_SZ     FIX-unsg (  2) = 0
59 _ParentID_OF     FIX-unsg (  4) = 104
60 _ParentID_SZ     FIX-unsg (  2) = 27
61 Title_OF         FIX-unsg (  4) = 530
62 Title_SZ         FIX-unsg (  2) = 13
63 Ext_OF           FIX-unsg (  4) = 100
64 Ext_SZ           FIX-unsg (  2) = 4
65 Author_OF        FIX-unsg (  4) = 0
66 Author_SZ        FIX-unsg (  2) = 0
67 Subject_OF       FIX-unsg (  4) = 0
68 Subject_SZ       FIX-unsg (  2) = 0
69 Keywords_OF      FIX-unsg (  4) = 0
70 Keywords_SZ      FIX-unsg (  2) = 0
71 Comments_OF      FIX-unsg (  4) = 0
72 Comments_SZ      FIX-unsg (  2) = 0
73 Snippet_OF       FIX-unsg (  4) = 131
74 Snippet_SZ       FIX-unsg (  2) = 399
75 URL_OF           FIX-unsg (  4) = 0
76 URL_SZ           FIX-unsg (  2) = 0
77 MIME-Type_OF     FIX-unsg (  4) = 32
78 MIME-Type_SZ     FIX-unsg (  2) = 10
79 Language_OF      FIX-unsg (  4) = 0
80 Language_SZ      FIX-unsg (  2) = 0
81 Encoding_OF      FIX-unsg (  4) = 0
```

```
82 Encoding_SZ       FIX-unsg (  2) = 0
83 _Created_OF       FIX-unsg (  4) = 68
84 _Created_SZ       FIX-unsg (  2) = 26
85 _Modified_OF      FIX-unsg (  4) = 42
86 _Modified_SZ      FIX-unsg (  2) = 26
```

Viewing Words in the Index

You can use the didump utility to see words in a collection. The didump utility includes information about the occurrences of words in a single partition. Output can be viewed on the screen or redirected to a file. If words have been auto-stopped, you may find a value of 0 in the first four columns of the index. If you are concerned that a particular word might occur frequently and wish to check the status, you can use the didump utility and specify the word.

For instance, from the coll a collection, you could use the didump to see the occurrences of the word "the" in the word indexes which are stored under the parts directory.

```
didump -pattern the 00000000.did

      Word  Size  Doc   Word
      THE   8     1     1
      The   98    5     27
      the   605   6     195
```

BUILDING COLLECTIONS WITH THE COMMAND-LINE SPIDER: VSPIDER

The optional command-line spider (**vspider**) provides additional indexing options and greater control than is available in the Information Server Indexing Manager. Following are listed some of the more common commands for building a collection or maintaining a collection using vspider. Remember that vspider is an optional product. It is very helpful for routine collection building, for instance at night, when you can set it to run in chron, without having to issue indexing commands directly.

All commands need to be entered on one line:

Creating a New Collection

```
vspider -collection mycoll.clm -style verity/s97is/locale/english/styles
-start http://www.yoursite.com
```

Re-indexing a Collection

```
vspider -collection mycoll.clm
```

Using domain and exclude options

```
vspider -collection mycoll.clm  -style verity/s97is/locale/english/styles
-start http://www.yoursite.com  -exclude  *secrets*  -domain verity.com
```

Using the proxy option:

```
vspider -collection mycoll_2.clm  -style verity/s97is/locale/english/styles
        -start  http://www.verity.com -proxy  proxysvr: 8010
```

Using an authorization file (for password challenge)

```
vspider     - collection mycoll_3.clm
            -style verity/s97is/locale/english/styles
            -start  http:// www.verity.secure.com/index.htm
            -auth authfile.txt
```

VSPIDER Options

```
-style       Style directory (if metaweb add .clm to collection name

-start       Starting URL or URL's for the spider to follow
             -start http://www.verity.com   http://www…

-collection  Path to where the collection is to be created
```

Indexing options:

```
-allold          forces reparsing of all documents in the collection

-cgiok           allows indexing of URL's containing the ? symbol to permit
                 indexing CGI's.  Default is to exclude these.

-domain          limits indexing to specified domains


-exclude         excludes files matching a specified expression
                 exclude *bat *exe

-host            limits indexing to specified hosts.

-include         includes files matching a specified expression
                 include  *rpt

-maxdocsize      maximum size of document to index  (default is 1 megabyte).
                 Enter values in kilobytes

-mimeexclude     specifies mime types to be excluded. -mimeexclude  text/html

-mimeinclude     default is to include all mime types - specify only one

-nofollow        disables the following of href commands

-norobo          specifies that any robots.txt files encountered are ignored.
                 The default is to honor all robots.txt files.


-noupdate        disables updating of any documents already in the
                 collection

-pathlen         allows you to specify  how far down a URL or directory path
                 the spider should go.  The web host name is not included
```

	(www.verity.com:9000/) but elements following it are. Each slash counts like this/that/then/what would be -pathen 4.
-submitsize	specifies the number of documents to be submitted at one time (default is 50)
-topicset	specifies the topicset to be used for indexing the collection
-unlimited	specifies no limits are to be placed on the spider, if neither -host or- domain is specified. The default is to limit based on the host of first URL listed. Also removes the 1 megabyte maxdocsize default unless specified.
-verbose	displays summary information for each URL or file accessed. Indicates indexed, ignored or failed.
-debug	displays debugging-level messages (more than verbose)
-trace	provides highest level of information about indexing.
-auth	specifies the name of the authorization file for sites requiring a password.
-casesen	makes processing case-sensitive so the spider separately keeps differing only in case.
-charmap	specifies character map to use
-common	specifies path to VDK home.
-datefmt	specifies import format for date Y-M-D M-D-Y etc.
-help	displays syntax options
-language	specifies local language
-jobs	maximum number of sockets to allocate (default is ten)
-msgdb	specifies the path to the ind.msg message database
-noindex	specifies to submit, but not index (used with mkvdk-persist)
-noproxy	specifies the host listed should bypass the proxy server (wildcards ok *.mysite.com)
-proxy	specifies the host &port for the proxy server
-purge	purges the contents of the collection specified
-repair	repairs a damaged collection
-temp	specifies directory for temporary files (default is /tmp)

BUILDING FILE-BASED COLLECTIONS WITH MKVDK

The mkvdk utility is an all-purpose collection maintenance tool. When you use this utility, you supply command-line options to tell mkvdk what you want to do. For example, when you are creating a new collection you will provide different options than when you want to do collection maintenance. On-line help with mkvdk is available by entering: **`mkvdk -help`**. Your Collection Building Reference Guide includes a full description of each of the values. A complete list with brief description is included in the Additional Materials section of this workbook. Here are some you will use often.

General Processing Options

`-collection colls/memos`	The collection name to be created or opened
`-synch`	Performs work immediately, rather than in the background as processing time permits
`-about`	Shows information about the collection (description and date last modified)
`-datapath ../db`	Specifies a path to use to find documents. All relative paths will be relative to this setting. If not used, mkvdk looks for the documents next to the collection directory.

Collection Creation Options

`-create`	Specifies the collection named in the -collection option is to be created. If it already exists, mkvdk exits and does not write over it.
`-style styles/wysiwyg`	Only used in conjunction with the -create option, this option specifies the path to your selected style directory.
`-description "My Memos"`	Sets the collection's description. Place in quotes.
`-words`	Builds the word list for each of the individual partitions in the collections

Document Processing Options

`-extract`	Indicates to parse field values as defined in the style.tde file.
`-insert`	Adds documents to the collection (this is the default for mkvdk).
`-update`	Add documents to the collection, replacing all previous information about the specified documents.
`-delete`	Deletes documents from the collection

Messaging Options

`-quiet`	Overrides standard -outlevel (which is set to 15 to show fatal, error, warning and status messages) and only shows fatal and error messages.

The process of building a collection with mkvdk is accomplished in three steps:

- Build the collection style
- Locate the documents to be indexed and put their names into a flist
- Run the mkvdk utility that creates the collection from the style files, flist, and other parameters required

The easiest way to create a new style is to borrow from an existing style and make any necessary changes or additions. The second step, building an flist, can happen in one of many ways. You could just type the names and paths directly into an ASCII file and use that as your flist, or you can use external utility programs that will help you build the list automatically. The third step requires that you run the mkvdk utility with the command-line arguments to create the kind of collection you desire. Housekeeping and optimization are handled automatically as the collection is built.

PRACTICE LAB #2
CREATING COLLECTIONS

1. In the IS97 folder, you will find a **colls** directory (the master directory containing the collections you will create), and a **docs** directory containing sub-directories of sample documents.

2. Open Netscape and return to the Information Server's Welcome page (you should have a bookmark (http://train1.verity.com:7000/search97secure/s97a.vta?). Click the "Configure Server" button to return to the Quick Start. If you don't already have a bookmark for this page, now would be a good time to make one. From the Quick Start, select the Index Manager from the menu bar.

3. Create a new collection called **"legal"** and use "Miscellaneous Legal Documents" as the description. Remember the collection has to be created empty first, then you add an indexing task to actually perform the indexing. Enter the full path to the collection directory: **c:\is97\colls\legal**

4. In URLs to index, enter: **file://c:\is97\docs\doc1**

   Click the Submit button to start the indexing task. When your indexing task has been submitted, click on the View Current Indexing Tasks icon.

5. The Current Indexing Tasks form displays current tasks, even if finished. You will notice that there are several indexing tasks submitted. For each indexing task you will see the name of the collection, the path to where the collection is created, and the status. While indexing is performing, a working icon is displayed. When finished the icon will change. Click on the Collection Manager icon from the menu bar.

6. From the Collection Manager, click on the View Collection List icon to see all of the completed collections. You should now see your new collection.

7. To search against your new collection, click the Quick Start icon, choose Search Your Website, and select your new collection from the pop-down list. Enter a word or phrase and click the Search button. If you do not have results in your collection, please notify your instructor.

Building Collections Using VSPIDER

8. Click Start and open an MS-DOS Command for some practice building collections with **vspider**. This indexer can be used to index documents managed by a remote http server (with appropriate licensing). You will use the command line to build your collection using the main style directory, as indicated in the example below. Substitute your "site of

choice" for the **-start** command, and build your collection. The vspider indexer has already been added to your environment path which means, if you are at the c:\> prompt, you will not have to enter the full path to vspider.exe. If you are in another directory, enter the full path c:\search97_nti31\admin\vspider.exe. Enter the following command on one line:

c:\search97_nti31\admin\vspider.exe
-collection c:\is97\colls\mysite.clm
-style c:\search97\s97is\locale\english\styles
-start http://www.yoursite.com ***(choose a small site to index for yoursite)***

Browse an Existing Collection

9. View the information that is contained in your new collection using the browse utility. To do this, you will need to move to the parts directory under your collection (**c:\is97\colls\mysite\html\parts**). Type **dir** to view the partition contents. You can use the browse utility to view document attribute information contained in any of the files that end with the **.ddd** extension. Choose one of these now and substitute it in the command shown below. Don't be surprised when the information flies by quickly in this window, in a moment we will have you redirect the output to a file. Enter the following command to view a partition:

c:\search97_nti31\bin\browse.exe 00000001.ddd

A short menu will appear. At this point you can press Enter and the first record will be presented.

```
?) help
 q) quit
 c) Number of entries in field
 _) Toggle viewing fields beginning with '_'
 v) Toggle viewing selected fields
##) Display all fields in specified record number
Dispatch/Compound field options:
 n) No dispatch
 d) Dispatch
 s) Dispatch as stream
Action (? for help):
```

There is a lot of information contained in the **.ddd**. As this is a very basic utility you will have to anticipate the commands as you redirect the output to a file. Enter the last command again (use up arrow to recall the previous command) adding the redirect as shown below:

c:\search97_nti31\bin\browse.exe 00000001.ddd > browse.txt

Press Enter twice (first time to execute browse and second time for the first record). Now type quit and your prompt will return. Enter dir to see the file that was created and use the (type) command to view the contents: type browse.txt

Your log file should look like this.

```
browse - Verity, Inc. Version 1.5.6 (_nti31, Nov 18 1996)
BROWSE OPTIONS
  ?) help
  q) quit
  c) Number of entries in field
  _) Toggle viewing fields beginning with '_'
  v) Toggle viewing selected fields
 ##) Display all fields in specified record number
Dispatch/Compound field options:
  n) No dispatch
  d) Dispatch
  s) Dispatch as stream
Action (? for help):

Record number: 0
0  _DDFLAG          FIX-unsg (  1) = 0x00
1  _DDVALUE         VAR-text (  0) =
2  _DDVALUE_OF      FIX-unsg (  4) = 0
3  _DDVALUE_SZ      FIX-unsg (  2) = 0
(followed by the rest of the field information…)
```

Updating and Deleting Documents in the Collection

10. Create another collection for practice on updating and deleting documents. Using **vspider**, build this collection as shown below and note that we have turned on advanced logging (called **-trace**) to learn more about the indexing process. Again, redirect the output to a file for your review. Enter the following command on one line:

    c:\search97_nti31\admin\vspider.exe -collection
    c:\is97\colls\personal.clm
    -style c:\search97\s97is\locale\english\styles
    -start c:\is97\docs\doc5 -trace > build.txt

11. Another utility that ships with the Information Server is **rcvdk**. This utility provides a little command line retrieval client. Review the information that is contained in your **personal** collection using the rcvdk utility, by moving to the bin directory (**c:\search97_nti31\bin**) which contains all the utilities you've been working with, and entering the following command.

    rcvdk.exe c:\is97\colls\personal.clm (press Enter)

The following message will appear:

Attaching to collection: c:\is97\colls\personal.clm
Successfully attached to 1 collection.
Type 'help' for a list of commands.
RC>

Type help to view the list of commands, and practice with these a bit on your collection.

12. Because information does not remain static, vspider comes with utilities which allow the updating of information already in the collection. This exercise will give you the opportunity to delete a document and appropriately update the collection, and then to change a document, and again see the update process. Remember that your "personal" collection was created using the documents in the doc5 directory (c**:\is97\docs\doc5**). Move to the **doc5** directory and delete the file called "Upside.doc." Run **vspider** to update the collection as shown below:

c:\search97_nti31\admin\vspider.exe -collection
c:\is97\colls\personal.clm

13. Using what you already learned about rcvdk, take a look at the collection and note the document has been deleted.

14. Move back to the **doc5** directory (**c:\is97\docs\doc5**). Open the "Scan.doc" file and change the date of the document to today's date and add your name. Save your changes and run vspider once more to pick up your changes:

c:\search97_nti31\admin\vspider.exe -collection
c:\is97\colls\personal.clm

15. Using rcvdk, search for your name.

16. Add some of your collections to your server (and remember to reference your collections at the .clm level. Do you remember why?

INDEXING TOPICS AGAINST A COLLECTOIN

1. In order to take advantage of search performance benefits provided by topics, you must first create a compiled topicset of your outline file, then index the compiled topicset against a collection. Open the IS97 directory and explore the topics directory. This directory already includes a news.otl file you will use to create a compiled topic set. There is a batch file called Bld_top.bat that is already set up to execute the mktopics.exe utility on the news.otl file or you can use the mktopics.exe at the command line.

2. Open the news.otl file using Wordpad or Notepad. This file contains many different topics. Take a moment to review the different topics and their structure. Also, notice the **control: 1** at the top of this file. This keyword indicates that this file will be parsed when read by the Search97 compiler. For this reason, format is important.

3. Now you're ready to compile your news.otl file. You may choose to use the mktopics.exe utility at the command line or execute the batch file (bld_top.bat). To run it from the command line, use the mktopics.exe utility, which we have copied into this directory (it originates in the bin directory). At the command prompt, type the following:

   mktopics.exe –topicset c:\is97\topics\news –outline c:\is97\topics\news.otl

   To use the batch file, enter the following: **bld_top.bat**

   Once the topicset has compiled, you will have a sub-directory called news in the topics directory.

4. Create a new collection called "misc" using the documents in the c:\is97\docs\doc7 directory, and indexing them using the compiled news topicset. At the command prompt, type the following:

   vspider –topicset c:\is97\topics\news –collection c:\is97\colls\misc.clm
   -style c:\is97\styles –start c:\is97\docs\doc7

5. When the indexing is complete, and before launching the Information Server, you will need to update your inetsrch.ini file to add the directory path to your topicset. To do so, move to c:\search97_nti31\bin directory and open the inetsrch.ini file using Wordpad. You will see the [Common] area towards the top of this file, enter the path to your compiled topic set as shown below:

   [Common]
   Topicset=c:\is97\topics\news
   topichome=c:\search97\common
   ProductLongName=SEARCH'97 Information Server V3.1

ProductShortName=Information Server
PatchLevel=10
ProductHomeLink=http://www.verity.com
VTopicScriptName=/search97cgi/s97_cgi.exe

Save your edits, then exit inetsrch.ini file.

6. Changes made to the inetsrch.ini file require restarting the server to read in new changes to the configuration file. To stop and restart the server, select Services from the Control Panel. Scroll down the list of services and click on Search97 IS Admin to highlight it, then click the "Stop" button to stop the service, and then "Start" to restart the service. Close the Services dialog and launch the Information Server.

7. From the Collection Manager import your new collection **misc**.

8. Using the default search page under Quick Start, select the **misc** collection from the pop-down list, and enter the topic **politics** in the search box. Click the Search button.

9. Open several of the documents in the results list and notice the highlighted words include terms contained in the politics topic and not only the word politics.

ENABLING SEARCH AT THE SERVER

One of the most important things you can do to ensure a successful application is to present information to your users, about how to search and how to work with results. Information applications are more about this than anything else and it is up to you to know how your data can best be used and to support a variety of user experience levels in your applications. Knowing how to query effectively will help you design information systems that make using the Verity Query Language easy.

EXPLORING THE QUERY LANGUAGE

The value of a query language is seen in two areas: how easy is it for a novice to ask for information and get good results and how rich is the language for knowledge workers to precisely target specific results?

For the novice, the query language provides defaults

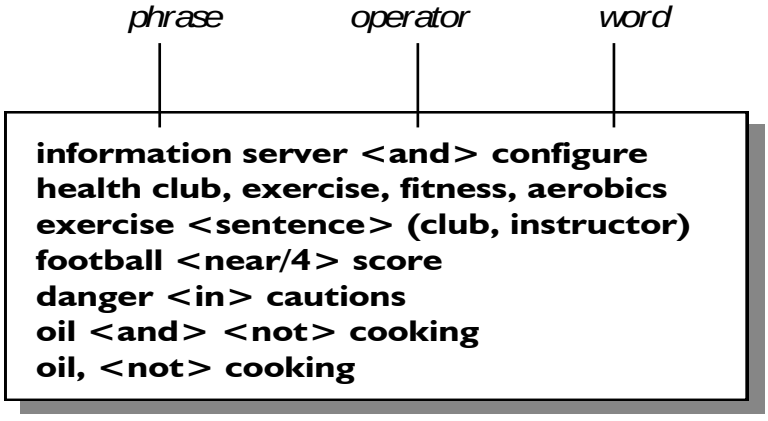
- standard variant endings
- density tie-breakers
- accrue on groups of words
- ranking by score with highest first

For the knowledge-worker, the query language provides

- a rich set of operators and modifiers
- parenthetic representation of complex ideas
- weighting of terms or groups of terms
- topics

WHAT IS A QUERY?

A *query* is simply the criteria you provide for performing a search. When you create queries, you can combine words, phrases, fields and topics with operators and modifiers to direct which documents will be selected and how they will be ordered on the results list



SYNTAX ALTERNATIVES

When you use `simple syntax` (the default), the query is interpreted with a broad focus. Your searches are case-insensitive and the STEM operator is automatically applied to search words, by selecting root word and standard variant endings. The MANY modifier is applied for search words to score documents higher based on word density words are automatically interpreted as topics when a matching topic exists. Groups of words are evaluated using Verity's proprietary ACCRUE operator at the parent level, to specify selection of any of the words entered, but higher scoring for those documents containing additional occurrences of unique words

When you use `explicit syntax`, you instruct the engine about how the search is to be handled. There are shortcuts for some explicit syntax operators and modifiers:

-<WORD>film or "film"
-<STEM> film or 'film'
-<SOUNDEX> @film@

When you want to search the text of the documents, you may do so by:

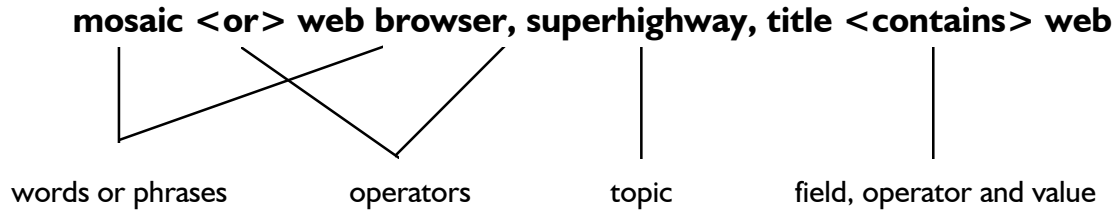
Query entering search words directly
information superhighway, online provider, internet, web

Topic entering the label specified for a group of words already defined or organized to represent a more complete subject
superhighway

QUERY COMPONENTS

Your queries can include any of these components:

- **WORDS** representing words in your documents (and found in the internal word index)
- **TOPICS** representing predefined groups of words and information about how the words relate to each other
- **FIELD VALUES** representing the value you will find in a particular field defined by the DBA
- **OPERATORS** to specify how a group of words should be searched



OPERATOR CLASSES

Evidence Operators search for words and can expand into a list of related search words, depending on the operator selected. **Proximity Operators** are used with groups of words to

define how closely they are related to each other. **Concept Operators** combine the meaning of search words to identify a concept in a document. **Relational Operators** are used with fields. **Boolean Operators** are used with topics and words to retrieve the elements you describe without operator precedence conflicts.

EVIDENCE OPERATORS

Evidence Operators represent your actual search words. Depending on the operator used, they may expand into a list of related search words:

Operator	Shortcut	Rule
<word> film	"film"	must locate an exact match on the word as entered (no variant endings to be included)
<stem> film	'film'	must locate a match on the root of the word and includes all standard variant endings (filming, filmed, films). This is the default.
<thesaurus> film		must search for all synonyms listed in the thesaurus for this word
<wildcard> tech*		must match the character string entered with selected variables *fil** *substitutes any characters for ** *fil?* *substitutes single letter for ?* *fi[ln]e* *substitutes specified letters in brackets* *film(s,ed)* *substitutes specified variables only as included in parentheses*

PROXIMITY OPERATORS

Proximity Operators define how closely words must be found near each other to qualify your subject. Proximity refinement often improves query results dramatically.

Operator	Shortcut	Rule
<phrase> nice job	nice job	must locate words next to each other, in the order defined
new <sentence> film		must locate words in the same sentence (any order)
hit <paragraph> film		must locate words in the same paragraph (any order)
weather <near> report		must locate words within 1000 words of each other and reflects proximity by score next to each other has highest score)

football <near/5>score	must locate words within the number of words specified by /n
danger <in> cautions	locates documents containing values in specific zones (like title). Only available for web-based collections or where DBA captures zones during indexing process.

CONCEPT OPERATORS

Concept Operators combine the meanings of a group of search words to identify a particular subject or concept in a document:

Operator	Shortcut	Rule
<accrue>	,	matching documents must contain at least one of the words entered but the more unique words, the better. This is the default.
<and>		matching documents must contain all of the words entered
<or>		matching documents must contain at least one of the words entered

RELATIONAL OPERATORS

Relational Operators are used with fields

Operator	Shortcut	Rule
<contains>		the string must be found within the field. This is the default.
<starts>		the field must start with this value
<ends>		the field must end with this value
<matches>		the field must contain a matching string
greater than	> (only)	the numeric field value must be greater than this number
less than	< (only)	the numeric field value must be less than this number
equals	= (only)	the numeric field value must be equal to this number

BOOLEAN OPERATORS

Boolean Operators are used with topics and words to retrieve the elements you describe without operator precedence conflicts

Operator	Shortcut	Rule
<any>		matching documents must contain at least one of the words entered
<all>		matching documents must contain all of the words entered

MODIFIERS

The behavior of operators can be modified or enhanced:

The **<NOT>** modifier is used to exclude documents

oil <AND><NOT> cooking

The **<MANY>** modifier is used to count the density of a word or phrase topic within a document

<MANY><WORD> earning

The **<CASE>** modifier is used to perform a case-sensitive retrieval on a word

<CASE><WORD> NeXT

The **<ORDER>** modifier is used to indicate the order of the words you have entered important

diver <ORDER><NEAR/5> kills <ORDER><NEAR/5> shark

THE IMPORTANCE OF TOPICS

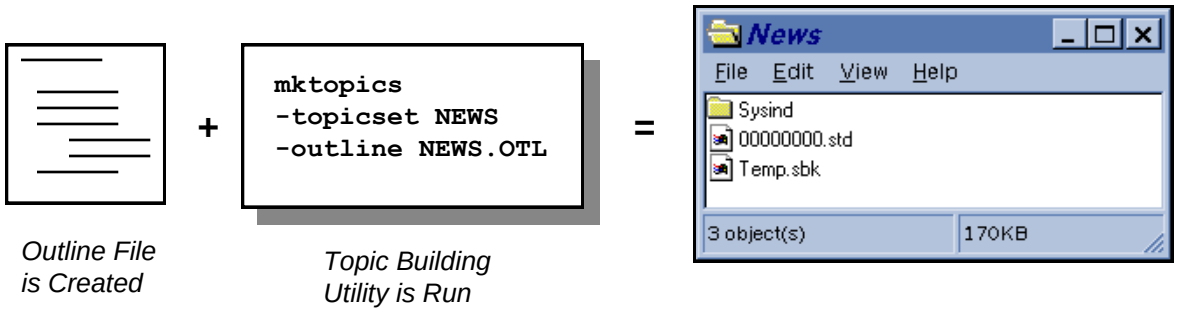
You can dramatically enhance the value of your applications and simplify searching for your users by incorporating topics. Topics represent proven knowledge about a particular subject. The more complex a subject, the greater the value of the topic. Differing points of view or levels of expertise can be addressed by your topics and searching is so much more effective as even novices benefit from expert knowledge captured in the topics

- Topics save money by eliminating redundant work and speeding you to the right information
- They can include words, phrases, field values, and the vast array of relationships between them
- Topics include all of the components of the query language (operators, modifiers, and weights)

- Topics can be indexed against collections. Searches using indexed topics are more than 30 times as fast as those without.

THE TOPIC BUILDING PROCESS

Topics are built by constructing the knowledge in a text file called an outline (OTL) file. Words are grouped in an information tree and operators, modifiers and weights are applied. Structure and syntax are important. The topics are then compiled by running the topic set creation utility, called mktopics, against the OTL text file. A special index is created which consists of the terms, relationships and when indexed against a collection of documents, the locations of matching documents.



Outline File is Created

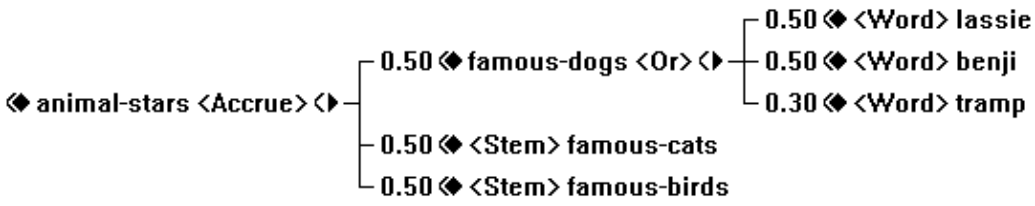
Topic Building Utility is Run

Topics are typically defined to serve as a knowledgebase for commonly asked questions and include definitions of relationships between words and rules for evaluating and scoring documents that are relevant to the search objective:

Evidence topics are search words, phrases or values that indicate the evidence or presence of your subject.

Branch topics represent subject categories within the general subject. Each branch of the tree defines a separate cluster of related subjects within the main topic. These are the subtopics

The **Root topic** (top-level topic) is a name or label which describes the general subject covered by the topic. This is the main topic.



Practice Lab #3
Practicing with Query Language

1. Select the Users Guide from the Quick Start (right under the Forms Wizard). Click on the link to Search'97 Search Tips. This section has been designed to help you understand the query language and how to write really effective queries. Go through this section and then using the documents you have available through your Information Server, try each of the operators. Remember that you need to know a little something about what is in the documents to make your search practice valuable, so look at them first.

2. For those operators having shortcuts, practice using the shortcuts in your query strings.

3. Add a more complex query by using parentheses to select groups of documents. For example:

   <ANY> (stock, trade, securities) <SENTENCE> <ANY> (market, exchange)

4. List some pre-defined queries that would be helpful at your site and be prepared to discuss how you could refine them for more precise results.

Build a Basic Topic

1 Create an outline file called **verity.otl** using wordpad and save it under the c:\is97\topics directory. Edit this file by creating a new topic called **verity**. Remember to include the keyword **control:1** at the top of the file.

Below are the components and search terms you will use to create your new topic. Organize each search word or phrase under the appropriate subtopic category:

Topic Name:	verity
Subtopic Names:	verity_products verity_services verity_technologies
Search Words:	search97, agent server, topic, search97 information server, topic agents, s97 information server, verity technical support, verity, verity consulting, document indexing, cd-web publisher, vspider, verity sales, document retrieval, verity training, search enabling, knowledgebase creation, push technology

When complete, save your edits and exit verity.otl file.

2. Create a compiled topicset on your verity.otl file. In the c:\is97\topics directory you will find a mktopics.exe utility you can use to compile your verity.otl file or simply update the Bld_top.bat file and execute this batch file.

At the command prompt type the following:

mktopics.exe –topicset c:\is97\topics\verity –outline c:\is97\topics\verity.otl

3. At this point, you will have a new compiled **verity topicset** directory. Edit the inetsrch.ini file to replace the current topicset with the new one that you have created. It is important to know that you can only have one topicset referenced in this configuration file .

   In the [Common] area towards the top of this file, enter the path to your new topicset as shown below:

   [Common]
   Topicset=c:\is97\topics\verity
   topichome=c:\search97\common
   ProductLongName=SEARCH'97 Information Server V3.1
   ProductShortName=Information Server
   PatchLevel=10
   ProductHomeLink=http://www.verity.com
   VTopicScriptName=/search97cgi/s97_cgi.exe

4. Save your edits, then exit inetsrch.ini file. Remember to stop and restart the Information Server.

5. Launch Information Server Admin, and use any of the default search forms to test your topics. s

INTRODUCTION TO SEARCHSCRIPT

- SEARCHScript Actions
- SEARCHScript Objects
- SEARCHScript Object Properties
- Operators and Functions

SEARCHScript is used to:

- Design forms and templates for use in Verity search applications
- Design HTML web page forms and select templates for
- Processing search requests
- Formatting search results
- Viewing documents
- Override configuration options and parameters defined in the Information Server Configuration File: inetsrch.ini

The SEARCHScript Language is typically embedded in a pair of <% control statement %> markers, like <% IF Document.Title = " " Then %>. SEARCHScript Language can also be placed outside of these control markers using substitution: Your Query $$QueryText matched $$DocsFound. The text inside of control markers is parsed and executed while text outside of the control markers is simply output

SEARCHScript is Verity's proprietary scripting language and is designed specifically for search-related functionality. It allows for alternatives in processing based on action types. SEARCHScript is typically defined in a separate, template file with an .HTS extension HTML pages reference the templates for processing. Substitutions are resolved and dynamic results are created according to the name-value pair guidelines. SEARCHScript can include conditional and iterative statements, as well as functions and comments like If -- Then -- Else – EndIf, For – EndFor and ForEach – EndFor.

```
<% -- Print a list of collections --%>

<% Foreach coll in Result.Collection %>
$$Coll
<% Endfor %>
```

Specify the operation you want to perform in structured statements. It can be part of a URL or it can be part of an HTML <FORM> tag.

```
http://sparky:7000:search97secure/s97a.vta?ACTION=SEARCH&
collection=News&QueryText="sports"

<FORM METHOD="POST" ACTION="$$Web.Scriptname">
<INPUT TYPE="hidden"  NAME="Action"  VALUE="Search"
```

Define actions in name-value pairs and assign variables, include files, etc. within the SEARCHScript control statement:

```
<INPUT TYPE="checkbox"  NAME="collection"
                        VALUE="mktg"> Product Information
<INPUT TYPE="text" SIZE=40 NAME="QueryText"  VALUE=" "
<INPUT TYPE="submit" VALUE="search"

<% variable = expression %>

<% QueryMode = "FreeText" %>
<% print(date($$doc._Modified,"$mm/$dd/$yy")) %>
<%-- #include file="identify.htm" --%>

<% if doc.mimeType=="pdf" %> doc.Pages
<% else %>
<% Val(doc.Size)/2048 %>
<% endif %>
```

SEARCHSCRIPT ACTIONS

SEARCH	Specifies query or search operations and results list formatting
FILTERSEARCH	Specifies query or search operations and results list formatting and allows you to specify a filtering template to perform query pre-processing
ADMIN	Performs configuration, indexing, or other administration tasks
FORMGEN	Generates search forms automatically, based on configuration parameters
DOCVIEW	Displays the contents of a document (with highlights) when selected from a results list

```
Enter words and phrases, separated by commas:<BR>

<INPUT TYPE="text" SIZE=40 NAME="QueryText"  VALUE="intranet"

<INPUT TYPE="checkbox" NAME="collection" VALUE="mktg"> Cool Stuff
<INPUT TYPE="checkbox" NAME="collection" VALUE="email"> Mail
<INPUT TYPE="submit" VALUE="Search"
<INPUT TYPE="reset"  VALUE="Clear"

http://sparky:7000/search97secure/s97a.vta?ACTION=SEARCH&
Collection=mktg&Collection=email&QueryText="intranet"
```

SEARCHSCRIPT OBJECTS

There are dozens of object properties that use SEARCHScript actions. These properties are organized in classes

Object	Property Description
Cluster	Information about documents included in clusters
Collection	Information on configured collections
Config	Default configuration file settings (applicable to all actions)
Document	Information about a specific retrieved document
DocView	Information useful for creating optional, customized views of documents
Request	Name-value pair information for a requested action
Result	Information on current query and documents retrieved in the search

```
Web          Information on the server and client software (applicable to all
             actions)

             <% Request.QueryText = " " %>
             <% Request.QueryMode = "Boolean" %>
             <% Foreach coll in Result.Collection %>
             <% Foreach coll in Config.Collections %>
             <% Print (Web.ServerURL + Web.Scriptname) %>
             <% Print (Web.Cookie) %>
             Indexed into:  $$doc.CollectionName
             Found at: $$doc.ParentID

              <% If exists(Doc.Title) Then %>
                   $$Doc.Title
                 <% Else %>
                   Oops, No Title
                 <% EndIf %>
```

SAMPLE OBJECT PROPERTIES

The SEARCHScript Reference Guide provides the current list of all script actions, objects and properties. Here are just a few to give you an idea of how they work. Please refer to the Reference Guide for the complete list.

CLUSTER OBJECT PROPERTIES

Clustering provides a technique for automatically discovering the subtopics in a set of documents and grouping the documents by those subjects:

```
Documents            Array of documents for a given cluster
                     <% foreach Doc in Cluster.Documents %>
Index                Element number for clustering
                     <% foreach cluster in Result.Clusters %>
                     <TD> Cluster <% Cluster.Index +1 %> </TD>
                     <TD> Score:  $$Cluster.Score </TD>
KeywordSummary       The first 6 keywords, space separated, for each cluster
                     <% foreach cluster in Result.Clusters %>
                     <TD> Summary $$Cluster.KeywordSummary </TD>
Keywords             Array of all keywords for a cluster (PDF only)
NextPageURL          URL_HTML to retrieve the next page of clusters returned
                     from a search
PageNumber           Page number of the current result list Web page
```

COLLECTION OBJECT PROPERTIES

Collection objects provide information about collections managed by the Information Server

```
Name                 Short name of a collection or collection map (.clm) file
                     <% foreach coll in collections %>
                     <LI><INPUT TYPE="checkbox"
                     NAME="collection"  VALUE = "$$Coll.Alias"
                     CHECKED>$$Coll.Name  <% Endfor %>
Alias                The automatically assigned alias for the collection name
DefaultList          Specify if the collecction gets searched - even if
                     not specified on search form
                     <% foreach coll in collections %>
                     <% If (Coll.DefaultList = "True") Then %>
                     <LI><INPUT TYPE="checkbox"
```

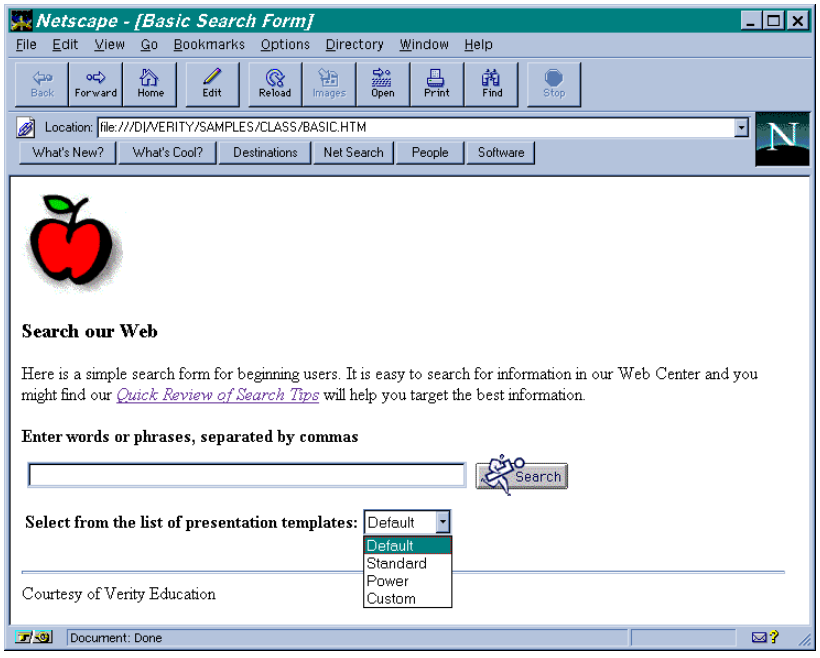
```
                    NAME="collection"  VALUE = "$$Coll.Alias"
                    CHECKED>$$Coll.Name
                    <% Endfor %>
Description         Information stored in the collection description
                    field
Path                Drive and directory path to the collection
State               Current state of the collection (enabled or
                    disabled) as specified in the inetsrch.ini file
                    <% If (Coll.State = "Enabled") Then %>
```

Customizing Searches and Results

Verity provides templates to give you ideas about how to customize your search forms. You can copy these existing pages, as a starting point and then customize them to fit your specific needs. The search form can be located anywhere on your system, but keep in mind that results are built dynamically and these forms must be located in whatever template directory is named in the inetsrch.ini file.

BUILDING YOUR OWN SIMPLE SEARCH FORM

This simple search form works well for new users in its simplicity yet provides powerful choices for viewing results.



Here is the HTML that builds the form above.

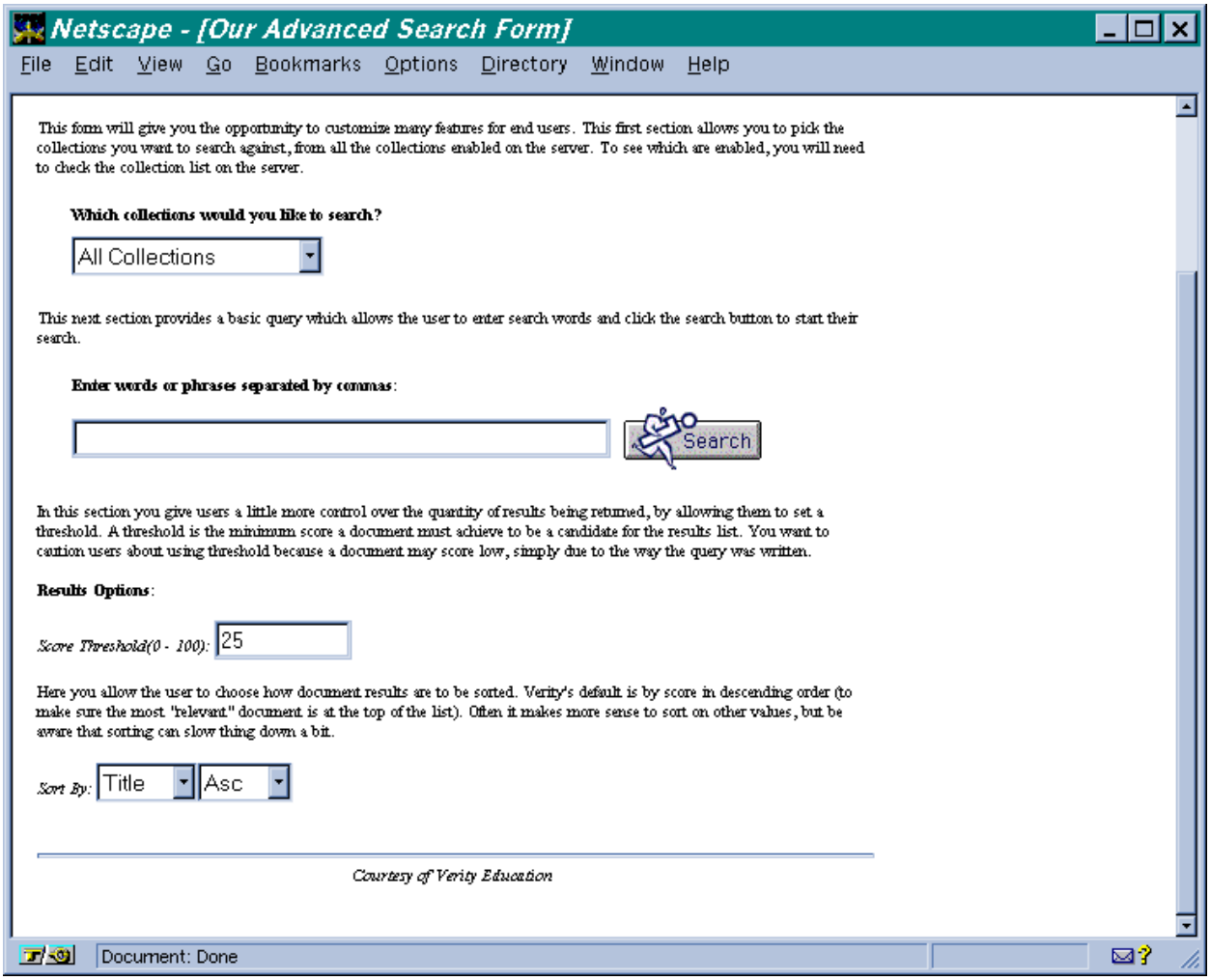
```
<HTML>
<HEAD><TITLE>Basic Search Form</TITLE></HEAD>
<BODY bgcolor=ffffff>
<form method="POST" action="/search97cgi/vtopic.exe">
<INPUT TYPE="hidden" NAME="Action" VALUE="Search">
```

```
<INPUT TYPE="hidden" NAME="SearchPage"
VALUE="/search97/samples/forms/srchdemo.htm">

<img src="apple.gif">
<H3>Search our Web</H3>
Here is a simple search form for beginning users.
It is easy to search for information in our Web Center
and you might find our
<A HREF="hottop1.htm">
<I>Quick Review of Search Tips</I></A> will
help you target the best information.<BR>
<P>
<b>Enter words or phrases, separated by commas</b>
<table>
<tr>
    <td><INPUT NAME="queryText" size=55 VALUE=""></td>
    <td><INPUT TYPE="image" SRC="search.gif" NAME="SEARCH-97" ALT="Search"
BORDER=0></td>
</tr>
</table>

<table border=0>
<tr>
<td>
<b>Select from the list of presentation templates:</b></td>
<td>
<select name="ResultTemplate">
<option value="basres.hts" selected>Default
<option value="stndard.hts">Standard
<option value="power.hts">Power
<option value="rbasic.hts">Custom
</select>
</td>
</tr>
</table>

<br>
<HR>

Courtesy of Verity Education

</body>
</html>
```

ADDING TIPS

It's a really good idea to include search tips for your users, especially if you expect to have regular "visitors" who have little experience with writing queries. You are welcome to link to any of the information points on the Verity web site and will find helpful information at the Verity Education site. Check out http://www.verity.com/educ.

PROVIDING ADDITIONAL FEATURES IN YOUR SEARCH FORMS



And here is the HTML that builds these choices:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
<HTML>
<HEAD>
<TITLE>Our Advanced Search Form</TITLE>
<META NAME="GENERATOR" CONTENT="Mozilla/3.0Gold (Win95; U) [Netscape]">
</HEAD>
<BODY BGCOLOR="#FFFFFF">
<TABLE WIDTH="520" >
<TR>
<TD>
<P>FORM METHOD="GET" ACTION="/search97cgi/vtopic.exe"><INPUT TYPE="hidden"
NAME="action" VALUE="FilterSearch"><INPUT TYPE="hidden" NAME="AdminImagePath"
VALUE="/search97admimg/"><INPUT TYPE="hidden" NAME="AdminScriptName"
VALUE="/search97cgiadm/vadmc.exe"><INPUT TYPE="hidden" NAME="AdminUserGuide"
VALUE="/search97/doc/user/"><INPUT TYPE="hidden" NAME="AdminSearchScriptGuide"
VALUE="/search97/doc/srchscr/"><INPUT TYPE=HIDDEN NAME="ServerKey"
VALUE="Primary"><INPUT TYPE="hidden" NAME="resulttemplate"
VALUE="results.hts"><INPUT TYPE="hidden" NAME="maxDocs" VALUE="100"><INPUT
TYPE="hidden" NAME="filter" VALUE="srchflt.hts"><INPUT TYPE="hidden"
```

```
NAME="command" VALUE="GetMenu"><CENTER><IMG SRC="apple.gif" HEIGHT=92
WIDTH=96></CENTER>
<HR><BR>
This form will give you the opportunity to customize many features
for end users. This first section allows you to pick the collections you
want to search against, from all the collections enabled on the server.
To see which are enabled, you will need to check the collection list on
the server. </P>
<TABLE>
<TR>
<TD WIDTH=10> </TD>

<TD COLSPAN=2><B>Which collections would you like to search?</B></TD>
</TR>
<TR>
<TD WIDTH=10> </TD>
<TD VALIGN=MIDDLE><SELECT NAME=collname><OPTION VALUE=ALL>All Collections
<OPTION VALUE=C1>S'97 Demo Collection <OPTION VALUE=Coll2>MyColl
</SELECT></TD>
</TR>
</TABLE>

<P>
This next section provides a basic query which allows the user to enter
search words and click the search button to start their search. </P>
<TABLE>
<TR>
<TD WIDTH=10> </TD>
<TD COLSPAN=2><B>Enter words or phrases separated by commas:</B></TD>
</TR>
<TR>
<TD WIDTH=10> </TD>
<TD VALIGN=MIDDLE><INPUT NAME="QueryText" SIZE=45 VALUE=""></TD>
<TD VALIGN=MIDDLE><INPUT type=image IMG SRC="srchbtn.gif" BORDER=0></TD>
</TR>
</TABLE>

<P>In this section you give users a little more control over the quantity of
results being returned, by allowing them to set a threshold.  A threshold is
the minimum score a document must achieve to be a candidate for the results
list.  You want to caution users about using threshold because a document may
score low, simply due to the way the query was written.   </P>

<P><B>Results Options:</B> </P>

<P><I>Score Threshold(0 - 100): </I><INPUT TYPE="text" SIZE=10
NAME="ResultCount" VALUE="25"></P>

<P>Here you allow the user to choose how document results are to be sorted.
Verity's default is by score in descending order (to make sure the most
"relevant" document is at the top of the list).   Often it makes more sense to
sort on other values, but be aware that sorting can slow thing down a bit.</P>

<P><I>Sort By:</I> <SELECT NAME="SortField"><OPTION VALUE="Score">Score
<OPTION SELECTED VALUE="Title">Title <OPTION VALUE="URL">URL </SELECT><SELECT
NAME="SortOrder"><OPTION VALUE="Desc">Desc
<OPTION SELECTED VALUE="Asc">Asc </SELECT></P>

<CENTER><P><BR>
```

```
</FORM>
<HR><B><I>Courtesy of Verity Education</I></B> </P></CENTER>
</TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

WHERE CAN I FIND MORE SAMPLES?

Each product in the Search'97 Application Suite has its own templates. For the Information Server, there are examples in the samples directory. Browse the sub-directories to find ideas for the functionality you would like to implement. Also check out the Verity website (demos) to get ideas implemented by other Verity search-enabled sites.

Practice Lab #4-A
Building Your Application

1. Review the search and result forms created by others by going to the Verity website, clicking on Demos, and then selecting Partner and Customer Demos. Each of these shows a search enabled web page in action. As you work through these note how others provide help tips, topics to choose from, predefined queries and collection selection. Take time to review the source, but keep in mind some of these pages may be using the older TIS version and have not upgraded to SEARCH'97 yet.

2. You are now ready to begin developing your search forms. The first task will be to customize a few basic templates in the Forms Wizard. This is the most logical way to get started with search forms - and the easiest. To help you get started, we have created a special html document (in class samples) called searchme.htm.

   To ready the environment for your development, open wordpad (which you will use to edit html files) and move it to the left margin on your screen (not fully expanded). Next, open Netscape (again, not fully expanded) and move it toward the right of your screen so that you can easily click back and forth between the two windows for testing.

3. Use File --> Open and select "searchme.htm" from the c:\is97\samples\template directory. Now, under Netscape select the bookmark for "Index of Samples." Choose searchme.htm and create a bookmark to this form as you will be using it a lot during this lab. Note that this page has a few starting images, color samples, etc., to help you get started.

4. Launch your Information Server and choose the Forms Wizard. Enter your company name and select the "simple" theme. Preview what this is to look like and then click Back to return to the Forms Wizard. Now click the apply button. The Forms Wizard

generates a URL that you will be adding to your "searchme" form. Using the right mouse button, highlight the URL and choose the option to “copy the link location.”

5. Move to the Wordpad window that is displaying searchme.htm. You will need to paste the URL that was copied directly before the text that it will link to and will need to add a few commands to turn it into a link (as shown below). Make sure there are no hard returns or spaces within the link (which has been broken below to fit on this page. We will do this in two steps.

   Paste the URL in the appropriate spot

   ```
   http://train7:7000/search97cgi/s97_cgi.exe?
   Action=FormGen&ServerKey=Primary&Template=smplsrcp.hts
   I want to do a simple search that presents basic results.
   ```

   Add html commands (shown in bold and italics) to complete this as a link

   ```
   <A HREF="http://train7:7000/search97cgi/s97_cgi.exe?
   Action=FormGen&ServerKey=Primary&Template=smplsrcp.hts">
   I want to do a simple search that presents basic results.</A>
   ```

6. Review the URL above and note the name of the search template that will be used. Write the name here: _____

7. Test your link by reloading searchme.htm and clicking on the link. You should see that you are now producing the same form that the Wizard produced as a dynamic search form.

8. Enter the query: `agents, topics` and the click search. Notice that you do not see what template is being used to format results in the URL. Go to View --> Document Source and notice that the template is named in the comment at the top of the result list as shown below.

   ```
   <!--- Template: smplrslp.hts --->
   <HTML>
   <HEAD>
   <TITLE>SEARCH'97 Information Server V3.0 - Simple Results List</TITLE>
   </HEAD>
   ```

9. Using your words and the images in this directory or copied from other sources, customize these two forms - the search form and the result form. Be sure to test them. Remember you will open the templates in wordpad, make edits and save, then test in Netscape to see your changes.

   It is easy to copy images from your site by highlighting them and using the right mouse button to “copy image location.” Be sure you save this in the c:\is97\samples\template directory so the server will be able to find them.

10. Go back to the Forms Wizard. Select the Power theme and follow the same approach to build and customize a power search form, linking it to your "searchme" page. Again, customize the search and results forms.

11. In Lab 2 you created topics and added them to your Information Server. Now we will use two of those topics to do automatic searching, but first please return to your Information Server and import the collection called news (under is97\colls). This collection has some news oriented documents that you can use for searches.

    Return to your searchme.htm form. Notice that there are two graphics on this form – one for sports and the other for politics. You will create a link that runs the appropriate topic, when someone click on either of these images.

12. Using the example below, add links to these images that run a search and pass the query. Notice that the topics are enclosed in curly braces {sports} as this is required to differentiate the use of a topic when being called from a template or HTML form. This is not required by users, who simply need to enter the word on a search form. Again, make sure there are no breaks or spaces in the links, even though you see them below. Be sure to substitute your server name.

```
<A HREF="http://train1:7000/search97cgi/s97_cgi.exe?Action=Search
&ResultTemplate=power.hts&QueryText={sports}">
<IMG SRC="/samples/sports.gif" border=0 HEIGHT=100 WIDTH=100></A>
```

13. Again, test your link and then go back and do the same for the political topic {politics}.

14. Take a few minutes to review the SEARCHScript in the .HTS template files and the resolved SEARCHScript you see when you view the source. Notice the many features that are available on the power form. Look at how the collections are built, how choices are given, etc.

15. Finally, go back to the samples directory and review the SEARCHScript that was used to create ksrchsite.htm. Here is the file that has all of the cookie information and processing. Edit this file using wordpad to reflect your server information (i.e., change train1:7000 to your server id). Be sure to use replace on the Edit menu to do this as there are six links in this file that need changing. Give it a try. You may want to incorporate some of these features in your application.

Practice Lab #4-B
Building Custom Search and Result Forms

1. You are now ready to plan your basic search form. By now you should have a pretty good idea about the features you will provide for your novice users. You need to customize both search and results forms. Remember that you will be presenting your choices to the rest of the class and will need to describe why you chose your particular approach, the key features you have included, what you had to do to enable those choices, what you found easy to implement and what was more difficult. Prior to class, we created a special samples directory which includes all the templates and images used by the Information Server, as well as other general examples you can use or build on.

   You should have a bookmark for "Index of Samples" already. If not, please enter this URL (substituting your server information) and create a bookmark. You will be coming back to this area many times during this lab.

   `http://train1:7000/samples/`

2. Click on basic.htm. This form is a good one for novices because it provides very basic search functionality "at a glance" and a link to more detailed information. Because the option to select collections is not available, all collections will be searched.

   Review the list of presentation templates. Each one formats results in a different way. Choose "Simple Listing" and enter the word "agent" in the query box. Click the Search button.

3. Review the SEARCHScript for this basic (basic.htm) searcher. Identify the following SEARCHScript components on the form that follows.

   a. The command that sends the action to the server
   b. Building the query box
   c. Running the "Simple Listing" result (and what is the name of this file?)
   d. Running the "About Document" result (and what is the name of this file?)
   e. Running the "Details" result (and what is the name of this file?)

BASIC.HTM

```
<HTML>
<HEAD><TITLE>Basic Search Form (Basic.HTM)</TITLE></HEAD>
<BODY bgcolor=ffffff>

<FORM method=POST action="/s97is.vts">
<INPUT TYPE="hidden" NAME="Action" VALUE="Search">

<center><img src="smapple.gif" height=40 width=42>
<H3>Search Your Personal Library</H3></center>
<FONT FACE="arial"> <FONT SIZE=-1> <FONT COLOR="#000080">
Welcome to your personal library.  It is really easy to search for information
by simply entering
words that identify your interest.  For example, if you are interested in
finding documents about
job openings at Verity, you would enter:  <FONT FACE="courier"> <FONT
COLOR="#FF0001">
Verity AND job postings. </FONT> </FONT><BR>
We've included a <A HREF="hottop1.htm">
<I>Quick Review of Search Tips</I></A> to help
you learn how to target the best information. <BR> </FONT></FONT></FONT>

<P>
<b><FONT FACE="arial"> <FONT SIZE=-1> <FONT COLOR="#FF0001">
Enter words or phrases, separated by commas:</b>
<table>
<tr>
    <td><INPUT NAME="queryText" size=55 VALUE=""></td>
    <td><INPUT TYPE="image" SRC="/samples/search.gif" NAME="SEARCH-97"
ALT="Search" BORDER=0></td>
</tr>
</table>

<table border=0>
<tr>
<td>
</FONT></FONT></FONT>
<FONT FACE="arial"> <FONT SIZE=-1> <FONT COLOR="#000080">

<b>Select from the list of presentation templates:</b></td>
<td>
<select name="ResultTemplate">
<option value="rstndard.hts" Selected>Simple Listing
<option value="rpower.hts">Display Info About Document
<option value="rbasic.hts">Detailed with Content Snippet
</select>
</td>
</tr>
</table>
</FONT></FONT></FONT>
<br>

<HR>
<td><FONT FACE="arial"> <FONT SIZE=-1> <FONT COLOR="#FF0001">
Here are some examples of searches: </FONT></FONT></FONT>

<table>
```

```
<tr>
<td><FONT FACE="courier"> <FONT SIZE=-2> <FONT COLOR="#000080">Verity,
products, features
</FONT></FONT></FONT> </td>
<td> </td><td> </td>
<td><FONT FACE="arial"> <FONT SIZE=-1> <FONT COLOR="#FF0001">
Finds documents with any of the words; those with more unique query words rank
highest    </FONT></FONT></FONT> </td> </td> </tr> <tr>
<td><FONT FACE="courier"> <FONT SIZE=-2> <FONT COLOR="#000080">Verity OR
Cognisoft    </FONT></FONT></FONT> </td>   <td> </td><td> </td>
<td><FONT FACE="arial"> <FONT SIZE=-1> <FONT COLOR="#FF0001">
Finds documents containing either word; number of hits (any word) determine
ranking       </FONT></FONT></FONT> </td>  </td>  <tr>
<td><FONT FACE="courier"> <FONT SIZE=-2> <FONT COLOR="#000080">Verity AND
Intranet      </FONT></FONT></FONT> </td>     <td> </td><td> </td>
<td><FONT FACE="arial"> <FONT SIZE=-1> <FONT COLOR="#FF0001">
Finds documents containing both of the words
</FONT></FONT></FONT> </td> <tr>
<td><FONT FACE="courier"> <FONT SIZE=-2> <FONT COLOR="#000080">Verity
&lt;IN&gt;
    TITLE
    </FONT></FONT></FONT> </td>
    <td> </td><td> </td>
    <td><FONT FACE="arial"> <FONT SIZE=-1> <FONT COLOR="#FF0001">
    Finds documents containing this word in the title zone
    </FONT></FONT></FONT> </td> </td>  </td> </table>

<table border=0>
<tr>
<td>
Courtesy of Verity Education

</body>
</html>
```

4. Review the SEARCHScript for the Simple Results List (rstndard.hts). Identify the following SEARCHScript components on the form that follows.

    a. The command that brings forward the previous query
    b. How many documents were found
    c. How many documents were searched
    d. Which documents (by rank number) are displayed on this page
    e. The creation of subsequent pages of results
    f. Putting the title on the result list with a link to the Verity highlighted document
    g. Putting the rank (the order of documents) on the result list
    h. Computing relevance value and showing an image to reflect its importance
    i. Computes the pages from the size
    j. Displays when the document was last updated (created or edited)
    k. Displays when the document was last indexed into a collection
    l. Reformatting of dates for both last indexed and last updated dates
    m. Substitution of "Title is Missing" if no title is found
    n. The name of the template being used RSTNDARD.HTS

```
<html>
<head>
<title>SEARCH'97 Information Server - Education's Standard Result
Template</title> </head>
<body    bgcolor="#ffffff"  text="#000000"  link="#0000ee" vlink="551a8b"
alink="ff0000">

<FORM method=POST action="$$web.scriptname">
<INPUT TYPE="hidden" NAME="Action" VALUE="Search">
<INPUT TYPE="hidden" NAME="ResultTemplate" VALUE="rstndard.hts">
<INPUT TYPE=HIDDEN NAME="ResultCount" VALUE="25">
</<% if exist( request.searchpage ) %>
<INPUT TYPE="hidden" NAME="searchpage" VALUE="$$request.searchpage">
<a href="$$request.searchpage"> <IMG src="$$web.ServerURL/samples/smapple.gif"
alt="[BACK TO SEARCH PAGE] "  BORDER=0 ALIGN=middle></a><p>
<% else %>
<a href="http://www.verity.com"> <CENTER> <IMG
src="$$web.ServerURL/samples/smapple.gif"
HEIGHT=40 WIDTH=42 alt="[Apples] "  BORDER=0 ALIGN=middle> </a><p>
<% endif %>

<b><FONT FACE="arial"><FONT SIZE=-1><FONT COLOR="#000080">Here is your last
query.
Would you like to change it based on the results you see?</b></FONT> </FONT>
</FONT> <table>
<tr><td><INPUT NAME="queryText" size=55 VALUE="$$request.queryText"> </td>
      <td><INPUT TYPE="image" SRC="/search97img/search.gif" NAME="SEARCH-97"
ALT="Search" BORDER=0></td>
  </tr>   </table> </FORM> </CENTER>

<b><FONT FACE="arial"><FONT SIZE=-2><FONT COLOR="#000080">Your query matched
<FONT COLOR=FF0001> <B>
$$DocsFound out of $$DocsSearched </B></FONT>documents in our library.
Documents $$DocsStart - $$DocsEnd are shown below, with best matches first. If
a row of numbers appears below, this means you have many pages of results
waiting behind the first group shown. Simply click on the next number to
proceed through matching documents.</FONT> </FONT> </FONT></b><dl>

<% if Count( result.pageurls ) > 1 %>
  <hr> <center>
      <% if result.prevpageurl then %>
            <A HREF="$$(result.prevpageurl)">[Prev]</A>
      <% endif %>

      <% for i = 1 to Count( result.pageurls ) %>
            <% if i != result.pagenumber %>
                  <A HREF="<% result.pageurls[i] %>">$$i</A>
            <% else %>
                  $$i
            <% endif %>
      <% endfor %>

      <% if result.nextpageurl then %>
                  <A HREF="$$(result.nextpageurl)">[Next]</A>
      <% endif %> </center>

<% endif %>
<hr>
```

```
<b><FONT FACE="arial"><FONT SIZE=-2><FONT COLOR="#000080">Color indicators are
provided to help you understand how well documents match your terms.  This
color scheme follows a standard traffic light, so while you have a green
light, you can be assured you have good results.  When the light changes to
yellow, you may need to proceed with more caution because these documents are
less relevant. <IMG src="$$web.ServerURL/samples/hot.gif"> means your document
scored very high. </FONT></FONT></FONT>
</b><P>

<% if Count( result.documents ) > 0 %>

  <table>
  <tr><font size =-1>
      <td><b><u>Rank</u></b></td>
      <td><b><u>Hot?</u></b></td>
      <td><b><u>Title</u></b></td>
      <td><b><u>Pages</u></b></td>
      <td><b><u>Last Updated</u></b></td>
      <td><b><u>Last Indexed</u></b></td>
      <td><b><u>Score</u></b></td>

  </tr> </font>

  <% foreach doc in result.documents %>
  <tr>
      <td><a href="$$doc.URL_HTML"><strong><center><font size =-1>
               $$doc.Rank</strong></center></font></td>
      <td valign=top><center>
          <img src="$$web.ServerURL/samples/<% Val( Right( $$doc.Score, 2 ) )
/ 10 %>ball.gif"></center></td>
      <td><a href="$$doc.URL_HTML"><strong><font size =-1>
      <% If exists(Doc.Title) then %>
      $$doc.Title
      <% Else %>
      Title Missing
      <% EndIf %>
      </strong></td>      <td valign=top>
      <center>
      <% if doc.mimeType == "pdf" %>
      doc.Pages
      <% else %>
         <% Val( doc.Size) / 2048 %>
      <% endif %>
      <td><font size =-1><strong>
      <%print(date($$doc._Modified,"$mm/$dd/$yy")) %></strong></font></td>
      <td><font size =-1><strong>
      <%print(date($$doc._Index_date,"$mm/$dd/$yy")) %></strong></font></td>

      <td><a href="$$doc.URL_HTML"><font size =-1>
      <strong>$$doc.Score</strong></font></td>
      </center>
      </td>
  </tr>
  <% endfor %>
  </table>

  <% if Count( result.pageurls ) > 1 %>
```

```
    <hr>
        <center>
        <% if result.prevpageurl then %>
              <A HREF="$$(result.prevpageurl)">[Prev]</A>
        <% endif %>
        <% for i = 1 to Count( result.pageurls ) %>
              <% if i != result.pagenumber %>
                    <A HREF="<% result.pageurls[i] %>">$$i</A>
              <% else %>
                    $$i
              <% endif %>
        <% endfor %>
        <% if result.nextpageurl then %>
                    <A HREF="$$(result.nextpageurl)">[Next]</A>
        <% endif %>
        </center>
  <% endif %>
<% else %>
  <center><b>No documents matched your query.</b></center>
<% endif %>
<I>Template: $$Template</I>
</body>
</html>
```

5. Take some time to test and review the differences in basic1.htm, basic2.htm and basic3.htm. Each one provides a unique set of features.

6. Keep in mind that you must always access these forms through your server in order to resolve the search script and produce results. Starting with the a1search.hts file, build your own basic form. Give the user a couple of choices that you like from the sample forms and then add at least 3 new things that are not on our search forms. Remember to refer to the SEARCHScript Documentation for ideas. Consider the alternatives you will want to present to your users. The new options you provide can also be used in your power search form. Use the list in the training guide for ideas about what you might want to include. Remember that the SEARCHScript only works when seen through the eyes of your Information Server and that you edit on the filesystem but test through the server

   http://train1:7000/samples/(*name of your file)*

Additional Materials

- Installation and configuration guidelines for Netscape Enterprise Server
- Configuring Multiple Servers

Installing the Information Server 3.1.0.

1. Prior to set up, an http server instance must be created for your Information Server. Take a minute to explore this http server instance. For center classes we use a Netscape Enterprise Server. Your instructor will provide server information for onsite courses. Launch your web browser and enter the appropriate URL:

   http://train1:8001 (Substitute the 1 after train with your student number)
   username: student
   password: learn

   Click on the **train1** button to access your web server. Then click on the "View Server Settings" from the Server Preferences list. Review the current settings, noting that a hostname and port has been provided. You will need this information during the installation process, so write it down.

   If you scroll down, you will notice that all the content settings refer to information about the Netscape Server and none reflect information about the Information Server. When you complete the installation, this area will be significantly updated to show many additional references to where key information about the server can be found.

2. Exit your web browser and any other Windows programs before beginning installation. Insert the SEARCH'97 Information Server CD in the drive. In a moment, the installer will be launched automatically. Continue from the "Welcome" dialog to the "Software License Agreement" dialog box. Please read the Software License Agreement and accept the terms by scrolling to the bottom of the agreement, click the "Yes" button to continue.

3. Choose to install the full version in the "Installations Options" dialog, click the "Next" button.

4. SEARCH'97 Information Server and Agent Server work in conjunction with an http server. In the "Web Server Installation" dialog box, choose Netscape Enterprise Server 3.0 as the http server, click the "Next" button to continue.

5. Leave the proxy host settings blank in the "Proxy Host Information" dialog box, and simply click the "Next" button to continue.

7. In the "Administration Information" dialog, enter "student" for the SEARCH'97 Information Server administrator user name and enter "learn" for the password. Verity's default port number (8989) is a re-direction port. After you install the Information Server, you will go back into the Web server to apply the new configuration settings. This creates the re-direction channel for processing requests through the Information Server. Click the "Next" button.

8. In the "Choose Destination Location" dialog, use the default target directory (c:\search97) for your installation, click the "Next" button.

9. In the "Select Program Folder" dialog, enter Verity in the Program Folder field as the destination for your program icons. Click "Next" button to continue.

10. Review the information in the "Start Copying Files" dialog. Click the "Next" button if settings are correct, or you may go back and change any setting necessary by clicking the "Back" button. Once you continue, the installation program will stop your http server and install the Verity software.

11. Enter the license information for your organization in the "Registration" dialog. This enables the functionality appropriate for your particular organization. (Your instructor will provide you with licensing information.)

12. The installation program will now update your system, restart the http server, and start the Verity services. The "Setup Complete" dialog box will finish the setup and prompt you to launch the readme file and SEARCH'97 Information Server Admin page. For now, uncheck both options and click the "Finish" button to exit out of set up program.

13. Finally, you must apply the changes (additional settings for Verity) in the Netscape Enterprise Server. To do so, launch your web browser and enter the following URL:

    http://train1:8001 (Substitute the 1 after train with your student number)
    username: student
    password: learn

    Click on the train1 button to access your http server. Netscape prompts you with a warning message: Manual edits not loaded. Click the "OK" button to continue, then click the "Apply" button on the upper right side of the menu bar. To apply your changes and load the latest configuration settins, click the "Load Configuration Files" button. When complete, a message will prompt you that recent config files have been loaded.

14. Now your ready to launch Search'97 Information Server Administration. To do so, enter the following URL:

    http://train1:7000/search97secure/s97a.vta?
    (Remember to substitute the 1 after train with your own student id number)
    username: student
    password: learn

    Click the "Use Frames" box on the SEARCH'97 Welcome page. One last setting you need to utdate is the license key for the Universal Spider, which will allows you to index multiple domains on different hosts or remote web sites. (Your instructor will provide you with a license key for the vspider.) Select the "Host Manager" button to bring up the server options page. Click the "License" icon from the menu bar on the left side. In the Update License page, check the "Universal Spider" box, enter the following license key, and then click the "Update" button to enable remote spidering:

Congratulations, you've just successfully installed Verity's SEARCH'97 Information Server and Agent Server!

SETTING UP MULTIPLE INSTANCES OF INFORMATION SERVER

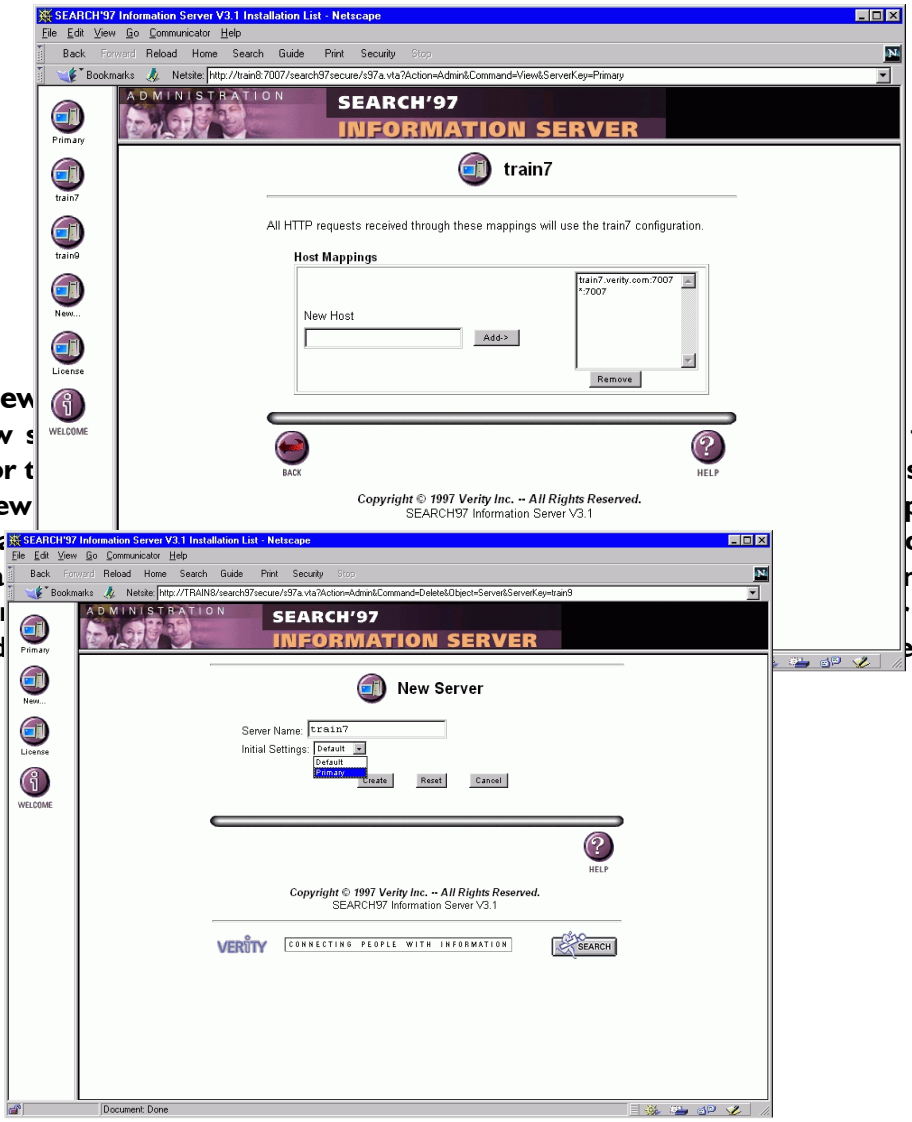
Information Server can be used in an environment where if you have multiple Web Servers (Netscape, MIIS, Apache) accessing one installation of Information Server, or if you need to create multiple instances of Information Server using one Web Server.

Information Server allows for multiple server instances by setting up different hosts, IP addresses, and ports in the Host Manager.

You can also set up multiple server instances for:

Security **- To limit access to a particular port to certain users. Each port will have access to specific collections, topicsets, and template files.**

Multiple Language Support **- Configure different IP addresses or ports to receive queries in different langauges.**

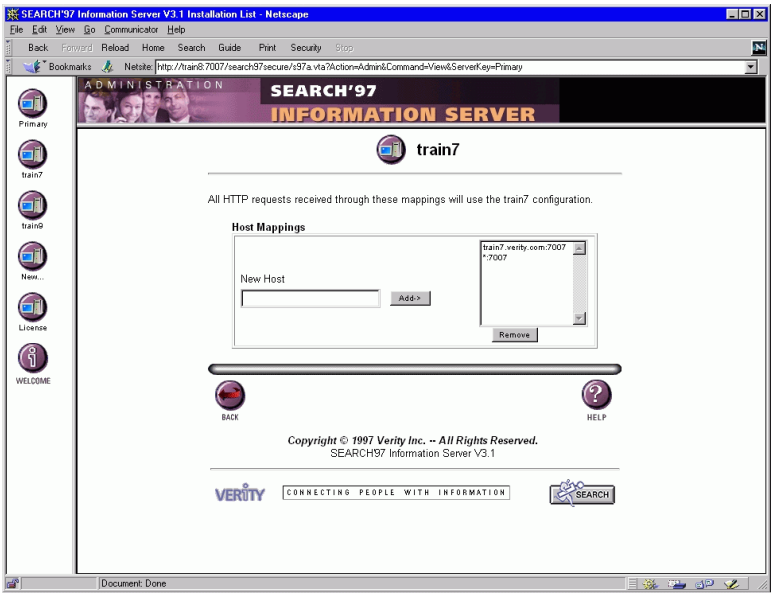


Setting Up a New
To create a new s **from the**
Welcome page or t **stance by**
selecting the New **provide a**
Server Name tha **ou have a**
choice to copy a **n without**
consideration fo **settings,**
and template d **es in the**
inetsrch.ini file).

Once the server instance is created, you will need to configure the settings under the Server Instance Options. Select Host Mappings, and enter a host and port number. This mapping must be set up with a port number that will also be used with your http server.

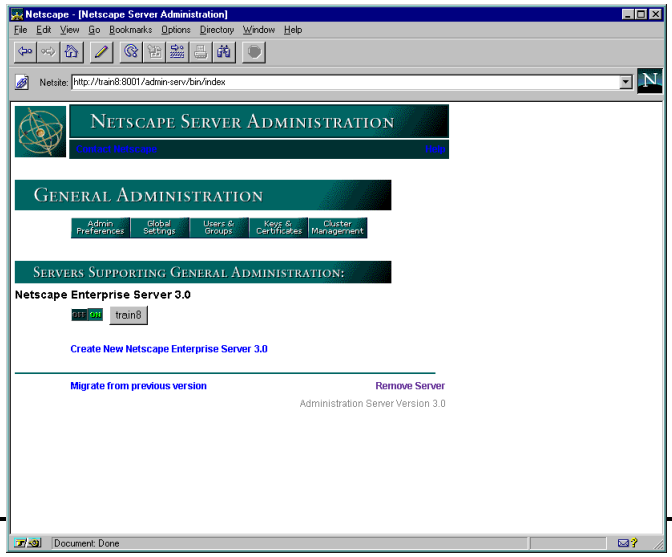
You can use any one of these host settings in the Information Server to map to your http server. Note, when you create an Information Server instance and assign it a new port number, that same port number must also be used in the http server instance.

Host name:7001	Host name.verity.com:7001
IP address:7001	*:7001

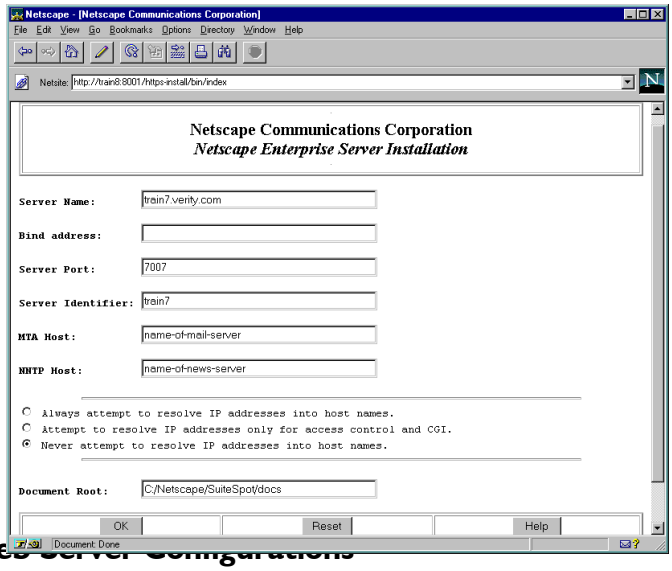


Creating a Netscape Webserver Instance

When you've completed the set up for a Search'97 Information Server instance, you will also need to create a new Netscape Webserver instance that maps to your new server instance. The Netscape Server Administration allows you to create, manage, and remove a Netscape Enterprise Server instance.



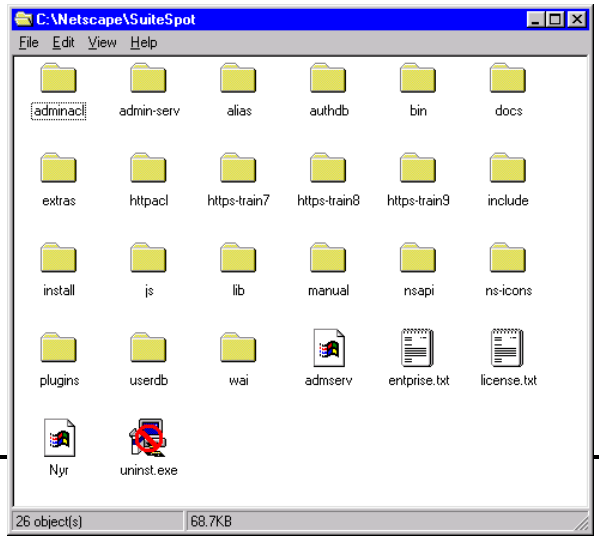
SELECT 'CREATE NEW NETSCAPE ENTERPRISE SERVER 3.0' AND A SERVER INSTALLATION PAGE WILL APPEAR, PROMPTING FOR NEW SETTINGS. THIS IS THE AREA WHERE YOU WILL PROVIDE SERVER INFORMATION FOR YOUR NEW INSTANCE. IN THE SERVER NAME FIELD YOU WOULD ENTER A HOST NAME AND DOMAIN, THE SERVER PORT NUMBER SHOULD BE A A NEW PORT NUMBER THAT IS NOT ALREADY BEING USED, AND FOR THE SERVER IDENTIFIER YOU WOULD INCLUDE A NAME THAT IDENTIFIES THAT HTTP SERVER

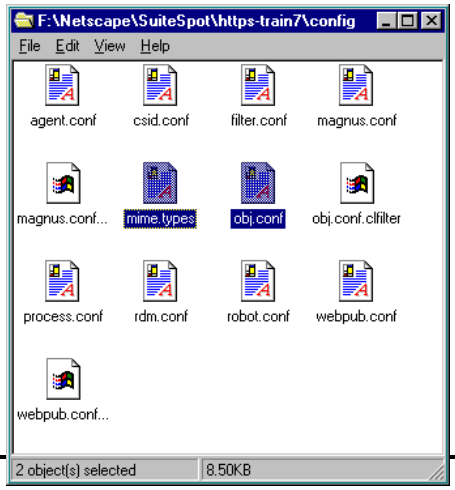


Updating the Web Server Configurations

There are two files that need to manually be updated to specify information about Search'97 Information Server. As you will notice, when yout create another Web server instance, it only specifies information (directories, and configuration files) about Netscape. There are no references about Search'97 Information Server.

The two configuration files obj.conf and mime.types that need to be copied or updated are located in the https directory of your new Web server install. For example, the picture below shows a new Web server install called https-train7. To the configuration files for https-train7, you would need to path to c:\netscape\suitespot\https-train7\config





To configure the obj.conf file, you would simply copy the original obj.conf file of your first Web server install for Search'97 Information Server. When you've copied obj.conf file into your new https server config directory, edit the file and change the line where it references the https server. See below example of obj.conf file, a bolded line indicates edits:

```
# Netscape Communications Corporation - obj.conf
# You can edit this file, but comments and formatting changes
# might be lost when the admin server makes changes.
# Use only forward slashes in pathnames--backslashes can cause
# problems. See the documentation for more information.

Init fn="flex-init" access="C:/Netscape/SuiteSpot/https-train1/logs/access" format.access="%Ses->client.ip% - %Req->vars.auth-user%
[%SYSDATE%] \"%Req->reqpb.clf-request%\" %Req->srvhdrs.clf-status% %Req->srvhdrs.content-length%"
Init fn="load-modules" shlib="c:/search97/_nti31/admin/s97a_ns2.dll" funcs="VerityTopicAdmin"
Init fn="load-modules" shlib="c:/search97/_nti31/bin/s97_ns2.dll" funcs="VerityTopicSearch"
Init fn="load-types" mime-types="mime.types"

<Object name="default">
NameTrans from="/search97securecgiadm" fn="pfx2dir" name="verity-secure-cgi" dir="c:/search97/_nti31/admin"
NameTrans from="/search97securecgi" fn="pfx2dir" name="verity-secure-cgi" dir="c:/search97/_nti31/bin"
NameTrans from="/search97secure" fn="pfx2dir" name="verity-secure" dir="/search97secure"
NameTrans from="/search97admimg" fn="pfx2dir" dir="c:/search97/s97is"
NameTrans from="/search97img" fn="pfx2dir" dir="c:/search97/s97is/locale/english/samples/images"
NameTrans from="/search97" fn="pfx2dir" dir="c:/search97/s97is/locale/english"
NameTrans from="/search97cgiadm" fn="pfx2dir" dir="c:/search97/_nti31/admin"
NameTrans from="/search97cgi" fn="pfx2dir" dir="c:/search97/_nti31/bin"
NameTrans fn="pfx2dir" from="/ns-icons" dir="C:/Netscape/SuiteSpot/ns-icons"
NameTrans fn="pfx2dir" from="/mc-icons" dir="C:/Netscape/SuiteSpot/ns-icons"
NameTrans fn="pfx2dir" from="/samples" dir="c:/is97/samples/template"
NameTrans fn="document-root" root="C:/Netscape/SuiteSpot/docs"
PathCheck fn="nt-uri-clean"
PathCheck fn="check-acl" acl="default"
PathCheck fn="find-pathinfo"
PathCheck fn="find-index" index-names="index.html,home.html"
ObjectType fn="type-by-extension"
ObjectType fn="force-type" type="text/plain"
Service fn="VerityTopicAdmin" method="(GET|HEAD|POST)" type="magnus-internal/vta"
Service fn="VerityTopicSearch" method="(GET|HEAD|POST)" type="magnus-internal/vts"
Service method="(GET|HEAD)" type="magnus-internal/imagemap" fn="imagemap"
Service method="(GET|HEAD)" type="magnus-internal/directory" fn="index-common"
Service method="(GET|HEAD)" type="*~magnus-internal/*" fn="send-file"
AddLog fn="flex-log" name="access"
</Object>

<Object name="verity-secure-cgi">
AuthTrans fn="basic-ncsa" auth-type="basic" userfile="c:/search97/userdb/verity.pwf"
PathCheck fn="require-auth" auth-type="basic" realm="Secure Access"
ObjectType fn="force-type" type="magnus-internal/cgi"
Service fn="send-cgi"
</Object>

<Object name="verity-secure">
AuthTrans fn="basic-ncsa" auth-type="basic" userfile="c:/search97/userdb/verity.pwf"
PathCheck fn="require-auth" auth-type="basic" realm="Secure Access"
</Object>
<Object ppath="c:/search97/_nti31/bin/*">
Service fn="send-cgi" type="magnus-internal/cgi"
</Object>
```

<Object ppath="c:/search97/_nti31/admin/*">
Service fn="send-cgi" type="magnus-internal/cgi"
</Object>

<Object name="cgi">
ObjectType fn="force-type" type="magnus-internal/cgi"
Service fn="send-cgi"
</Object>

See below example of mime.types file, a bolded line indicates the two lines that were copied into this file. Also, note that this is not a complete copy of the entire mime.types file.

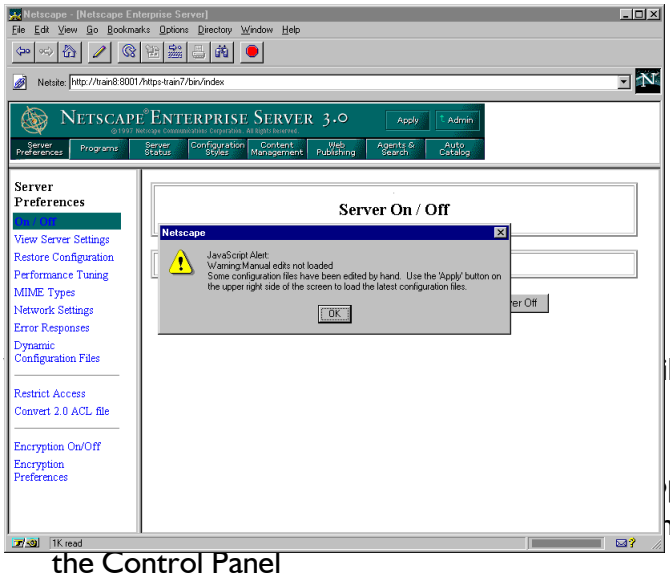
```
#--Netscape Communications Corporation MIME Information
# Do not delete the above line. It is used to identify the file type.

type=application/x-tex          exts=tex
type=application/x-texinfo      exts=texinfo,texi
type=application/x-timbuktu     exts=tbp
type=application/x-tkined       exts=tki,tkined
type=application/x-troff-man    exts=man
type=application/x-troff-me     exts=me
type=application/x-troff-ms     exts=ms
type=application/x-troff        exts=t,tr,roff
type=application/x-wais-source exts=src
type=application/zip            exts=zip

type=audio/basic                exts=au,snd
type=audio/echospeech           exts=es,esl
type=audio/midi                 exts=midi,mid
type=audio/x-aiff               exts=aif,aiff,aifc
#type=audio/x-midi               exts=midi,mid
type=audio/x-wav                exts=wav
type=audio/x-pn-realaudio       exts=ra,ram
type=audio/x-pac                exts=pac
type=audio/x-epac               exts=pae
type=audio/x-liveaudio          exts=lam

type=image/fif                  exts=fif
type=image/gif                  exts=gif
type=image/ief                  exts=ief
type=image/ifs                  exts=ifs
type=image/jpeg                 exts=jpeg,jpg,jpe,jfif,pjpeg,pjp

type=image/png                  exts=png
type=image/tiff                 exts=tiff,tif
type=image/vnd                  exts=dwg,svf
type=image/wavelet              exts=wi
type=image/bmp                  exts=bmp
type=image/x-photo-cd           exts=pcd
type=image/x-cmu-raster         exts=ras
type=image/x-portable-anymap    exts=pnm
type=image/x-portable-bitmap    exts=pbm
type=image/x-portable-graymap   exts=pgm
type=image/x-portable-pixmap    exts=ppm
type=image/x-rgb                exts=rgb
type=image/x-xbitmap            exts=xbm
type=image/x-xpixmap            exts=xpm
type=image/x-xwindowdump        exts=xwd

type=text/html                  exts=htm,html
type=text/plain                 exts=txt
type=text/richtext              exts=rtx
type=text/tab-separated-values exts=tsv
type=text/x-setext              exts=etx
type=text/x-speech              exts=talk

type=video/isivideo             exts=fvi
type=video/mpeg                 exts=mpeg,mpg,mpe,mpv,vbs,mpegv
type=video/x-mpeg2              exts=mpv2,mp2v
type=video/msvideo              exts=avi
type=video/quicktime            exts=qt,mov,moov
type=video/vivo                 exts=viv,vivo
```

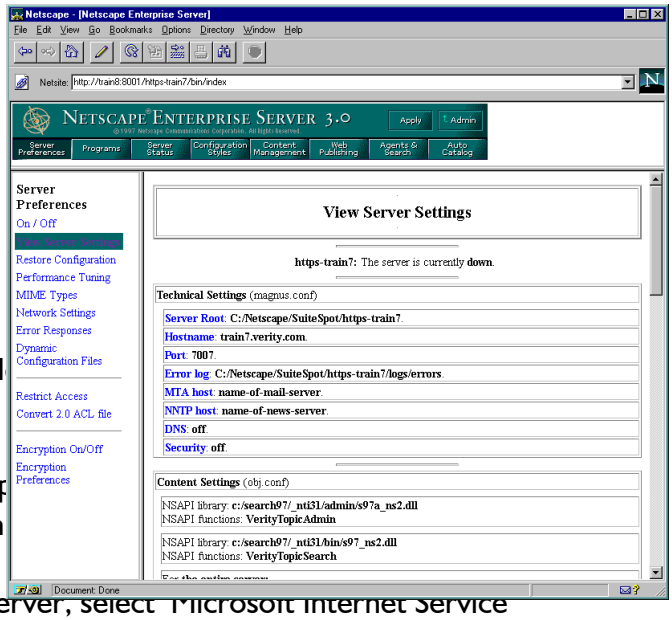
```
type=video/wavelet              exts=wv
#type=video/x-msvideo           exts=avi
type=video/x-sgi-movie          exts=movie

type=x-world/x-svr              exts=svr
type=x-world/x-vrml             exts=wrl
type=x-world/x-vrt              exts=vrt

type=x-conference/x-cooltalk    exts=ice

enc=x-gzip                      exts=gz
enc=x-compress                  exts=z
enc=x-uuencode                  exts=uu,uue

type=magnus-internal/vta exts=vta
type=magnus-internal/vts exts=vts
type=magnus-internal/imagemap    exts=map
type=magnus-internal/parsed-html exts=shtml
type=magnus-internal/cgi         exts=cgi,exe,bat

type=application/x-x509-ca-cert       exts=cacert
type=application/x-x509-server-cert   exts=scert
type=application/x-x509-user-cert     exts=ucert
```

type

Applying Changes to Web Server Instance

Any changes made to the configuration files of your Web server instance must be re-loaded in order to update configuration files. You can choose your new Web server instance from the Netscape Server Administration page to apply changes. Netscape will prompt you with a message that manual edits have been made to the configuration files and that you will need to click the 'Apply' button to load latest configurations





the Control Panel

- MIIS and Search'97 Information Server on NT Server, select 'Microsoft Internet Service Manager' from Programs
- Netscape Webserver and Search'97 Information Server on Solaris, start, stop, and restart script files are located in the install directory for both servers

Accessing Your New Server Instance

At the time you created a new server instance in the Search'97 Information Server, you provided it with a new port or IP address to access the Search'97 Information Server Administration

Enter the following URL in your Web Browser:

http//192.168.169.8:7007/search97secure/s97a.vta?

or

http://train1:7007/search97secure/s97a.vta?