

Internet'te Tarama Sistemlerinin Kurulması

Building Internet Search Engines

Mustafa Akgül*

Öz

Internet üzerinde var olan çeşitli tarama mekanizmaları, kullanıcılara birey ve kurum adreslerinden doküman adreslerine, istatistiklerden sözlüklere, kitap kataloglarından ürün fiyatlarına kadar bir yelpazede elektronik nesnelere aranıp bulunması ve sunulmasında yararlı olmaktadır. Hiyerarşik şekilde örgütlenen sanal kütüphanelerle birlikte, tarama mekanizmaları kullanıcıya bu çok büyük dağıtık kütüphane üzerinde yolunu bulmasına yardımcı olmaktadır. Bu makalede çok yaygın olarak kullanılan tarama motorlarının özellikle Unix ortamında kurulması ve çalıştırılması için yapılması gerekenler anlatılmaktadır.

Abstract

Internet search engines are powerful tools to find electronics objects such as addresses of individuals and institutions, documents, statistics of all kinds, dictionaries, catalogs, product information etc. This paper explains how to build and run some very common search engines on Unix platforms, so as to serve documents through the Web.

Giriş

Tüm dünyayı kapsayan bir iletişim ve elektronik nesne değişim ortamı olan Internet, aynı zamanda dünyanın en büyük kütüphanesidir. Dağıtık hesaplama ilkelerine ve istemci-sunucu (client-server) modeline göre çalışan Internet zaman ve mekânın öneminin azaldığı bir bütünlük oluşturmaktadır. Internet üzerindeki her kullanıcı var olan her şeye erişebilmekte ve sunucu bilgisayarı olan her kurum da dağıtıcıdaki elektronik nesnelere tüm Internet toplumu ile paylaşabilmektedir.

* Doç. Dr. Mustafa Akgül <akgul@bilkent.edu.tr> Bilkent Üniversitesi Endüstri Mühendisliği Bölümü Öğretim Üyesidir.

Henüz yeterince örgütlenmemiş, çok hızlı değişen ve sürekli büyüyen bu dağıtık kütüphanede nesnelere erişmek için birbiriyle ilintili iki kavram, 'sanal kütüphaneler' (virtual libraries) ve tarama mekanizmaları kullanıcıya yardımcı olmaktadır. Sunulacak nesnelere çoğaldığında tarama mekanizmaları kaçınılmaz olmaktadır.

İstemci/sunucu modelinde kullanıcı kendi bilgisayarında, bir Web tarayıcısı ile üzerinde dokümanlar bulunan bir bilgisayarı sorgulamaktadır. Sorgulama kullanıcının bilgisayarında bir pencereye kelimeler yazılarak yapılmaktadır. Sunucuya, aranan dokümanları tanımlayan anahtar kelimeler gönderilmekte, sunucu bu kelimelere uyan elektronik nesnelere bir program kanalıyla sorgulamakta ve sonuçları bir liste olarak istemci bilgisayara iletmektedir. Kullanıcı bunların arasından istediği dokümanları kendi bilgisayarına aktarmaktadır.

İstemci/sunucu modelinde verilen her hizmet belirli bir protokol ve 'port' üzerinden yapılmaktadır. Protokol iki bilgisayarın konuşmasını ve elektronik nesne değişimini düzenleyen kurallar manzumesidir. Her hizmete yönelik konuşma ise radyolardaki dalga boyuna benzetebileceğimiz bir port üzerinden olmaktadır. Her hizmet için standartlaşmış bir port vardır. Çeşitli nedenlerle bir hizmeti boş olan bir başka porttan sunmak da mümkündür. Örneğin, aynı makinede iki farklı Web sunucusu çalıştırmak için iki farklı port kullanılır.

Bu makalede "doküman" terimi tarama sonucunda kullanıcıya sunulan elektronik nesne anlamında kullanılmaktadır. Dolayısıyla doküman kendi başına bir metin, resim, ses, vs. dosyası olabileceği gibi, herhangi bir veri tabanının bir parçası da olabilir. Bu veri tabanı bir başka bilgisayarda çalışan bir veri tabanı motorunun denetimi altında olabileceği gibi (SQL temelliler), bir grup dosya içinde belirli şekilde ayrılmış kayıtlar da olabilir. Örneğin, bir dosyanın her bir satırı, boş satırlarla ayrılmış parçalar, herhangi bir ifadeyle ayrılabilen parçalar birer doküman olabilir.

"Tarama motoru" terimi taramayı yapan ve bunun alt yapısını oluşturan programlar için kullanılmaktadır. Bir tarama motoru bir veri tabanı motoru (Oracle, Access) ile etkileşime girebilir. Biz veri tabanının düz metin olarak sunucu bilgisayarda olduğu durumlara ağırlık vereceğiz.

Bu makalede anlatılan tarama motorları üç ana gruba ayrılabilir: 1) Doğrudan ilgili dosyalar üzerinde tarama yapanlar; **grep** ve **htgrep** gibi; 2) Sunucunun içinde var olan tarama motorları; **Gn** ve **Wn** gibi. 3) Tam metin dizinlemeye dayalı yöntemler. Bu programlar veri tabanı ya da dokümanları "dizinleme" dediğimiz bir ön hazırlıkla çeşitli yardımcı dosyalar oluşturmakta, tarama yapan programlar da bu yardımcı dosyaları kullanmaktadır. Taramaya tabi olan dosyaların büyük olduğu zamanlarda ve/veya taramanın hızla olması gerektiğinde bu tür tarama mekanizmaları gerekmektedir.

Sunucu-istemci modeli ile TCP/IP üzerinden dokümanları tarama ve erişim olanağı sağlayan araçların en önemlileri Wais, Isite, Glimpse ve Harvest'tir. Bunların ilki Thinking Machines, Apple, Dow Jones ve Wall Street Journal'ın ortak çabasıyla başlayan WAIS (Wide Area Information Server)'tir. Wais'te Z39.50 protokolünü esas alan sunucular birkaç sunucular dizinine (directory of servers) kayıt olurlar. Her wais sunucusu kendi taşıdığı tüm dokümanların dizinlerini tutar ve ağ üzerinden (LAN, WAN ya da uluslararası) bu dokümanları taramaya açar. Ayrıca dizinlenmiş dokümanları, Web, gopher üzerinden de sunmak mümkündür. Dizinlerin olduğu yerel bilgisayarda doğrudan istemci programlarla tarama yapmak mümkündür. Örneğin, kendi postanızı, makalelerinizi ve başka elektronik dokümanlarınızı wais ya da benzeri dizin/tarama araçlarıyla tarayabilirsiniz.

Tarama yapmak isteyen kullanıcı, istemci program kanalıyla, bir ya da daha fazla sunucuyu seçer ve bazı kelimeleri verip bu kelimelerle tarama yapılmasını ister. Her sunucu taramaları yapar, ve çıkan dokümanların listesini gönderir. Bu liste her doküman için, konusal yakınlık açısından isabet derecesini gösteren bir "başarı sayısı" da içerebilir.

Bu makalede esas olarak Unix ortamına göre tarama motorlarının kurulması ele alınmasına rağmen, verilen bilgiler Unix dışı ortamlarda da sistem yöneticilerinin işine yarayacaktır. Kavramların aynı olmasının yanı sıra, Unix dışı sistemlerde çalışan yazılımların çoğu zaten Unix kaynak kodlarının uyarlanması ile elde edilmiştir. Daha da önemlisi aynı tür konfigürasyon söz konusudur. Biz bu yazıda özellikle kütüphanecilere, ellerindeki dokümanları kolayca sunabilecekleri tarama mekanizmalarını nasıl kurabileceklerini anlatmayı amaçladık.

Burada yer darlığı nedeniyle açıklanamayan bazı detayları (Akgül, 1996a)'da bulabilirsiniz. Unix'e aşina olmayanlara Han (1993) ve Ayfer'i (1995) başlangıç için öneririz. Sistem yöneticileri için Pabrai (1993), Comer (1991), Frisch (1995); Internet servisleri için ise ek olarak Liu, et.al. (1994) ve Ford'u (1995) öneririz. Özellikle Web için önerilebilecek kaynaklar ise Akgül (1996b), Graham (1995), Lemay (1995) ve Morris'tir (1995). Kullanıcı açısından Internet'i anlatan kitaplar ise Akgül (1995), Ayfer (1996), Gaffin (1994), Korel (1992), Moore et al. (1994) ve Presno'dur (1996). Champine'in (1991) eserinde ise dağıtık hesaplama ve TCP/IP temelli bir üniversite bilgisayar sistemi anlatılmaktadır.

Program ve dosya adlarını yer yer **koyu**, yer yerde daktilo ya da Courier fontlarıyla yazdık. "Dizin" terimi hem isim, hem de fiil olarak 'index' karşılığı olarak kullanıldı. "Dizinleme" ile kastettiğimiz, tarama için esas metine ek olarak yeni yardımcı dosyalar üretilmesidir. Ortaya çıkan dosyaların tümüne de "dizin" diyoruz. Dizin terimini "directory" anlamına kullandığımızda parantez içinde mutlaka (directory) terimini ekledik. Web, gop-

her, ftp, html, wais gibi çok kullanılan sözcükler bu makalede küçük harflerle yazılmıştır.

Unix Ortamında Programları Kurmak İçin Öneriler

Unix ortamında programları kurmak (install) için bazı öneriler: önce README, INSTALL dosyalarını ve varsa başka dokümantasyonu okuyun. Bu dosyalar genellikle ne yapılması gerektiğini belirtir. Bazen 'configure', 'Configure' gibi programları çalıştırıp, sorulan sorulara cevap vermek gerekir. Sonra 'Makefile, conf.h, config.h' gibi bazı dosyalarda bazı düzeltme ve tercihler yapılır. Daha sonra 'make', 'make all', 'make install' gibi komutlar kullanılır. Program bir X11 uygulaması ise 'Imakefile' dosyası üzerinde bazı tercih ve düzeltmeler yaptıktan sonra 'xmkmf', 'xmkmf-a' ve daha sonra 'make Makefile', 'make Makefiles' ve 'make' gibi komutlar gerekebilir. Bu bilgi genellikle README/INSTALL dosyalarında vardır.

Unix ortamında programların davranışı, genellikle, çevresel değişkenler ve komut satırında verilen seçeneklerle değiştirilebilir. Derleme sırasında 'hard coded' bazı dizin (directory)/dosya parametreleri ise 'soft link' mekanizması ile düzenlenebilir.

Programları birden fazla ortam için derlemek zorundaysanız; X11 ailesinden 'indir' programını kullanabilirsiniz. Bu program, gerekli dizinleri yaratır ama dosyaların 'soft link' ile bağlantısını sağlar. Biraz dikkatle, iki ayrı ortam için disk alanınızı artırmadan programları derleyebilirsiniz. GNU programlarında --srcdir seçeneğiyle de benzeri bir işi görmek mümkün olabilir. Aksi takdirde, bir ortamdan ötekine geçerken *make distclean* ya da benzeri komutlarla ciddi bir temizlik yapmalısınız.

Programları açarken izlenecek yol: Paketin içinde neler var; nasıl bir dizin (directory) yapısı var, ve dosyalar sağlam gelmiş mi sorusu için *gzcat paket.tar.gz | tar tvf -* komutunu girmek yeterli olacaktır. *tar xvf -* ise dosyaların açılmasını sağlar; *v* (verbose) dosya isimlerinin ekrana yazılmasını sağlar.

Free Software Foundation GNU projesi adıyla Unix'le sınırlanmayan, kaynakları dağıtılan bir işletim sistemi geliştirmektedir. Bu proje kapsamında pek çok önemli yazılım ortaya çıkmıştır. Kurumların temel GNU araçlarını ve kardeşlerini (gcc, make, bison, flex, m4, perl, groff, python, expect, tcl/tk, safe-tcl, ghostscript, patch, wdiff, gzip gibi) ve TeX'i taşımalarıyla yarar vardır.

URL: Internet'te Nesne Adresleri

Internet'te sunulan nesne adresleri genellikle, protokol, makine adı ve o makinede söz konusu protokole göre erişim yolunu (path) içerir. Internet aracılığıyla erişilebilen dokümanların yerini tanımlayan evrensel adrese URL (Uniform Resource Locator) diyoruz. Genel olarak bir URL'i şöyle yazabiliriz:

```
URL: protokol://[[user[:passwd]@]dagitici-makina\
[:port] [/path-selection]
```

Burada [. .] olarak belirtilen kısım seçimli kısımdır. Aşağıda bazı URL örnekleri verilmektedir:

```
file://localhost/ftp/ftp/pub/INFO/Turkce/Internet/inet2.tex
ftp://ftp.bilkent.edu.tr/pub/INFO/Turkce/Internet/inet2.tex
gopher://gopher.bilkent.edu.tr:70/00/bilkent/archive/INFO/.../inet2.tex
http://www.bilkent.edu.tr/prv/ftp/INFO/Turkce/Internet/inet2.tex
wais://dagitici-makina:210/veritabani-adi
telnet://user@makina
news:news-grup
news:makale-numarasi@makina
news://makina/news-grup
ftp://user:passwd@makina
http://www.bilkent.edu.tr/cgi-bin/phf?Qname=caglar$Qtitle=&Qdepart-
ment
http://www2.infoseek.com/Titles?qt=turkey
http://query3.lycos.cs.cmu.edu/cgi-bin/pursuit?usenet+site
```

İlk dört URL Akgül'ün (1995) TeX formatındaki kopyasını belirtir. file: sadece 'ftp' diskini gören Bilkent'teki her makine için geçerlidir, diğerleri evrenselidir. Sondaki ftp://'li örnek de bir Web istemcisi ile kendi hesabınıza ftp yapmak için kullanabileceğiniz bir URL'dir. Şayet kullanılan port standart ise :port kısmına gerek yoktur. 'path-selection' kısmında söz konusu nesneye erişmek için gerekli yol belirtilir. Telnet'te 'user' kullanılması gereken kullanıcı adıdır. Bu durum genellikle kullanıcıya hatırlatılır ve kullanıcının bu bilgiyi girmesi beklenir. /cgi-bin/phf?.. URL'inde ise, 'phf' programına soru işaretinin (?) sağındakiler, çeşitli saha ve aldığı değer olarak iletilmiştir. phf telefon/e-posta adreslerinin tutulduğu veri tabanını sorgulayın 'ph' programına geçit veren bir programdır.

Grep Temelli Taramalar

Unix makinelerinde grep ve agrep gibi programlar bir grup dosya üzerinde kelime temelli tarama yapılmasına izin verirler. Tarama sonucunda ilgili dosyanın adı ve aranan ifadeye uyan satırlar kullanıcıya sunulur. Taranacak ifade . ? + * ^ \$ [] | \ karakterlerinin özel anlamı olduğu bir düzgün ifade (regular expression) olabilir. grep -i ise taramada küçük harf büyük harf farkını ortadan kaldırır.

Grep

Bir Web sunucusunda 'cgi-bin' dizinindeki programlar, shell, Perl vs. script'leri kullanıcıyla etkileşime girebilir; kullanıcıdan tarama için kelime isteyip, o kelime için yapılan tarama sonuçlarını kullanıcıya sunabilir. Gene bir Web sunucusu kullanıcıdan tarama için basit kelime girmesini istemenin dışında bazı karmaşık seçenekler sunabilir ve kullanıcıdan paragraflar dolusu girdi alabilir. Bunlar birbirine bağlıdır, ikisi de cgi-bin'den bir program kullanmak zorundadır.

Bir html dosyasında tarama olanağı olduğunu istemciye bildirmenin yolu <isindex> yapısının bulunmasıdır. Bunu gören istemci program, sorgulama penceresini/mekanizmasını alır ve sonucu sunucuya gönderir. Tarama yapıp sonucu gene istemciye sunmak sunucunun görevidir.

Bir URL'nin sonunda ? ve ?keyword bulunabilir. Daha doğrusu, istemci bazı anahtar kelimeleri sunucuya bu şekilde bir URL ile gönderir.

Belirli bir dosya üzerinden 'grep' taraması yapan bir script ise:

```
#!/bin/sh
echo Content-type: text/html
echo
if [ $# = 0 ]
then
  echo "<HEAD><TITLE>XYZ Telefon Taramasi \
    </TITLE>"
  echo "<ISINDEX>"
  echo "</HEAD> <BODY>"
  echo "<H1>XYZ Telefon Taramasi </H1>"
  echo " Tarama kutusuna istediginiz kelimeyi giriniz"
  echo " Bu taramada kucuk harf, Buyuk HARF farketmez"
  echo "</BODY>"
else
  echo "<HEAD><TITLE><b>=$*=</b>icin \
    Sonuclar:"
```

```

echo "&lt;/TITLE> &lt;/HEAD> &lt;BODY>"
echo "&lt;H1> \
      Tarama Sonucla-ri:&lt;b>=$*&lt;/b>&lt;/H1>"
echo "&lt;PRE>" \
      /usr/bin/grep -i "$*" /u/www/Webdocs/Personnel
echo "&lt;/PRE> &lt;/BODY>"
fi

```

Yukarıdaki script'te geçen `<`'nin `<` demek olduğunu ve `echo " .."` komutunun `"..."` ifadesini ekrana yazdığını belirtelim. Diyelim ki bu script'i `cgi-bin/grep-per` adıyla saklıyoruz ve modu `0755`. `http://sunucu/cgi-bin/grep-per` URL'ini gönderen bir istemciye `if .. else` arasındaki kısım gönderilir. İstemci o zaman bunun taranabilen bir dosya olduğunu anlayıp (ISINDEX) kullanıcıya tarama mekanizmasını sunacaktır. Kullanıcı istediği kelimeyi dolduracak; istemci de bunu sunucuya gönderecektir. Bu defa `else .. fi` geçerlik kazanacak, ve bu arada `grep -i "$*" /u/www/Webdocs/Personnel` işlemi sonucu oluşan çıktı da istemciye gönderilecektir. Şayet bu 'script'i kullanmak isterseniz, `/u/www/Webdocs/Personnel` kısmını değiştirmek yeterli olacaktır. Yalnız bu 'script'in güvenli olmadığını, kullanıcıdan gelen kelimeleri bir süzgeçten geçirmeden shell'e iletmenin tehlikeleri olduğunu belirtmek isteriz. Bu script'i Perl ile yapmak ve istemciden dönen kelimeleri bir güvenilirlik süzgecinden geçirmekte yarar vardır. Böyle bir Perl script örneği Bilkent arşivinden, Bilserv kanalıyla `alias_search` adıyla alınabilir.

Htgrep

Htgrep, Perl temelli bir tarama mekanizmasıdır. Perl4 ve Perl5 ile çalışır. Oscar Nierstrasz tarafından geliştirilmiş ve CUI W3 tarama mekanizmasında kullanılmaktadır. `PerlLib.tar.gz` ve `OscarScripts.tar.gz` paketlerini kullanır. Bunların bir dizinde (directory) açılması ve içindeki script'lerin `PERLLIB_INC` kısmının uyarlanması gerekir.

Tam bir örnek olarak Bilkent'te var olan `inet-all.html` dosyası üzerindeki taramaya bakalım. Yukarıda belirtilen dosyalara ek olarak taramayı yapan `/cgi-bin/inet-all` script'i:

```

#!/usr/local/bin/perl
# Configuration: give the location of the htgrep script
$HTGREP_LIB = "/akgul/etc/perl";
#location of the catalog file (cat.html) as relative
URL

```

```

$CAT_LOCATION = "/inet-yedek/inet-all/cat.html";
#----- do not change this -----
unshift(@INC,$HTGREP_LIB);
require("htgrep.pl");
&htgrep'settags($ENV{'PATH_INFO'});
&htgrep'settags($ENV{'QUERY_STRING'});
$htgrep'tags{'file'} = $CAT_LOCATION;
$htgrep'tags{'style'} = "dl";
&htgrep'doit;

```

Burada \$CAT_LOCATION Web'in başlangıcına bağlı olarak göreceli olarak verilmiş (yani URL olarak); \$HTGREP_LIB ise mutlak olarak verilmiş durumdadır.

Taranacak dokümanın olduğu dizinde cat.html, cat.hdr, ve cat.qry dosyaları var: cat.qry dosyasının script'i aşağıda verilmektedir:

```

<HEAD><TITLE>Inet-all -Search Result </TITLE></HEAD>
<BODY>
Bilkent'teki <a href="http://www.bilkent.edu.tr/inet
turkey/"> Internet Services in Turkey </a> dizininde
listelenen sunucular üzerinde <H1>inet-all tarama so-
nuclari</H1>
<HR>
<FORM ACTION="http://www.bilkent.edu.tr/cgi-bin/inet-
all">
Please enter a search word/pattern or provide a
<A HREF="http://www.bilkent.edu.tr/pl-regex.html">
Perl regular expression</A>:<P> Bir isim girin: <p>
<INPUT TYPE="submit" VALUE="Submit">
<INPUT NAME="isindex"> <P>
<B>NB:</B> Searches are case-insensitive.
Taramalarda küçük harf büyük harf farketmez.
</FORM>
<HR> Yeni servislerinizi
<a href="/inet-turkey/servis-ekle.html"> \
ekleyebilirsiniz</a>

```

cat.hdr ve cat.qry dosyaları tarama sonucunda kullanıcıya sunulan sayfanın baş kısmını oluşturur. Tarama hiçbir şey bulamazsa cat.hdr, bu- lursa cat.qry kullanılır, ama ikisi de aynı olabilir. Taranacak dosya ise cat.html olarak isimlendirilmiş ve her bir kayıt boş satırla ayrılmış du- rumdadır. Bu dosyada bulunan bağlantı bilgileri bir çapa (anchor) olarak okuyucuya sunulacaktır. Bilkent'te yansısı (mirror) sunulan CUI W3 kütüp-

hanesi 5Mbyte'ın üzerinde bir büyüklüğe sahiptir. inet-all için kullanılan cat.html'den birkaç satır aşağıda verilmektedir:

```
<a href=http://www.pcworld.com.tr/> PC World Turkiye </a>
Basili derginin elektronik kopyasi ve ilave sayfalar.
```

```
<a href=ftp://ftp.metu.edu.tr/> ODTU Archive </A>
Simtel, Cica, GNU, Linux, Networking vs yansilari ...
```

Sunucuda Kurulu Tarama Yöntemleri: GN ve WN

GN aynı zamanda hem Web hem de Gopher protokolüne göre hizmet verebilen bir sunucudur. WN ise sadece bir Web sunucusudur. GN ve WN'in en temel özelliği erişilen dokümanlar üzerindeki denetim kolaylığıdır. Dokümanlar teker teker adı uygun bir dosyada belirtilmeden erişime açılmazlar. Normal Web ve Gopher'da o dizindeki her nesneye erişebilirsiniz. Bazı tarama yetenekleri GN ve WN'de çok daha iyidir.

Şu anda gn-2.24 GN'in en son sürümüdür. Bu sürüm Bilkent arşivinden alınabilir. GN'in çalışmasını düzenleyen menu, .access, .cache dosyaları vardır. GN ve WN'in docs/ dizinlerinde (directory) yeterli dokümantasyon bulunabilir. GN'in örnek menu dosyaları bu dizinde bulunabilir. GN'nin 'root' dizininden başlayarak, erişim olan her dizine bir menu dosyası konur. Bu dosyanın içinde erişimine izin verilen dosya ve dizinlere ait bilgiler ve bu bilgilerin türleri tanımlanır. Sonra 'mkcache' programı çalıştırılır. '-r' opsiyonu da alabilen bu program mevcut dizinde (-r ile alt dizinlerde de) menu dosyasını okur ve cache dosyasını yazar. Sunucu cache dosyasına göre hizmet verir.

GN'in önemli özelliği dosyalar üzerinde mantıksal yapılar tanımlamaya izin verip, onlar üzerinde tarama yapmanıza olanak sağlamasıdır. Bu amaçla menu içinde iki parametre tanımlanır: Separator ve Section. 'Separator' söz konusu dosyayı bölümlere ayıran düzgün ifade (regular expression) tanımlamak için kullanılır, 'Section' ise sunulan mönüde her bölümün başlığını belirler. Örneğin, bir Unix makinesindeki e-posta mesajlarından oluşan bir dosya için Separator=^From_, (burada '_' boş bir karakteri belirtiyor) ve Section=^Subject: seçerek, her bir mesajı ayrı bir doküman olarak sunabiliriz. Dokümanların başlığını ise 'Subject:' satırının geri kalan kısmı oluşturacaktır. Bu tür dosyalar için Type=1, Path=1m/path-to-dosya parametreleri kullanılır. 'Separator' ve 'Section' parametreleri tanımlanmış bir dosya için Type=7 Path=7m/path-to-dosya seçenekleri konursa o zaman söz konusu dosya üzerinde tarama yapabiliriz. Tarama sonucunda 'Separator' ve 'Section'ın belirttiği birim dokümanlar sunulur. Bu

yöntem büyüklüğü 1 Mbyte'a kadar olan dosyalarda çalışır; daha büyük dosyalarda wais temelli yöntemlere gidilebilir.

Bu iki parametre tanımlanırken herhangi bir düzgün ifade kullanılabilir. Bilindiği gibi ^ satır başını, \$ ise satır sonunu, ^\$ ise boş satırları belirtir. ^[A-Z] bir satırdaki ilk karakterin A'dan Z'ye bir harf olmasını belirtir. Bir dosya üzerinde hem 1ⁱⁿ hem de 7ⁱⁿ tanımlandı ise, ikinci tanımda 'Separator' ve 'Section' yazılmasına gerek yoktur. Şayet aynı dosya üzerinde birden fazla yapı tanımlanacaksa, dosyanın Unix ln -s eski-ad yeni-ad komutuyla yeni bir görüntüsünü alıp, yeni görüntü için ayrı 'Separator' ve 'Section' tanımları yapılmalıdır. Böyle bir dosya için dosya-adi.cache isimli ek bir dosya gerekir. Bunu ise gn paketinin bir parçası olarak derlenen 'mkcache' programı yapar. Bu iki görüntü cache'leri ayırdetmeye yarar. Şayet kayıtlar ('record') arasında bir boş satır var ise; o zaman Separator=^\$ ve Section=\$^ yapabiliriz. Bu durumda, her boş satır yeni kaydın başladığını belirtir, ve boş satırdan sonra gelen tüm satır başlık olarak kullanıcıya sunulur.

Tarama İçin menu Örnekleri

Bilkent'teki GN altındaki "Internet Directory" dizinindeki (directory) menu den bazı örneklere bakalım:

```
Name=Hepsi Uzerinde Tarama
Path=7m/inet-dir/all.txt
Separator=^$
Section=$^
Type=7
```

```
Name>Listelerdeki isimler uzerinde tarama
Path=7m/inet-dir/listeden.txt
Separator=^
Section=^
Type=7
```

```
Name=Kisiler Dizini Uzerinde Tarama
Path=7m/inet-dir/birey.txt
Separator=^Ad:
Section=^Ad:
Type=7
```

Separator =^\$ ve **Section**=\$^ bir dosya içinde boş satırla tanımlanmış kısımları birer doküman olarak sunar. Dosyanın ilk satırında boş satır ol-

ması gerekir. Boş satırın altındaki satır, dokümanın başlığı olarak kullanıya sunulacaktır. İki tane boş satır arka arkaya gelirse, başlık olarak boş satır sunulur. Diğer bir deyişle fazla boş satır olmaması gerekir. Ve son satırın boş olması hatadır. Şayet 'Hepsi Üzerine Tarama' nesnesinde '7m' ve '7' de 7 yerine 1 kullanılsaydı; o zaman dokümanlar bir seçenekler mönüsü olarak sunulacaktı. Bunların ikisini bir arada yapmak mümkündür. İkinci örnekte, **Separator**=^ ve **Section**=^ ise, her satırın bir doküman olmasını sağlar. Bu ise `grep` taramasına özdeştir. Üçüncü örnekte ise, her doküman satır başında ilk üç harfi `Ad:` olan satırlarla belirlenmiştir. Boş satırların bir etkisi yoktur. Tarama sonucunda çıkan mönüde `Adi:` kısmı gözükmeyecektir. Bunların örneklerini Bilkent'te <http://gn.bilkent.edu.tr:7001> altında "Internet Hotel"de görmek mümkündür.

WN Web sunucusu tarama yetenekleri açısından çok zengindir. Ek bir konfigürasyona gerek olmadan başlıklar, anahtar kelimeler, alan (field), içerik (context), `grep` taramaları yapmak mümkündür. Anahtar kelime, ve alan için dizin temelinde bir konfigürasyon dosyasında taranacak bilgiyi koymak mümkündür. Örneğin bir dizindeki (directory) dosyalar üzerinde tarama yapmak için şu form yeterlidir:

```
<form method="GET" action="search=ontext">
<input name="query">
<input type="submit" value="Search"> <p> </form>
```

GN'de olduğu gibi bir dosyayı yapısal dokümanlara ayırmak mümkündür. Bu iş için `digest` programını kullanmak gerekir. `digest ^$ $^ foo` komutu `foo` dosyasından `foo.index.html` dosyasını yaratır. Bu yeni dosya GN'de olduğu gibi `foo` dosyasındaki yapısal dokümanları sunar. Kolay bir şekilde Web sayfalarına önceki, sonraki, tepe, ve tarama çapalarını (anchor) koymak mümkündür.

Wais

Wais'te sunulan nesne bir dosya olabileceği gibi, dosyanın belirli parçaları, dosya adı (ses, resim dosyaları için), paragraflar, satırlar v.s. olabilir. Bu tip programların üç parçasını sayabiliriz: `dizinleyici`, `sunucu` ve `istemci`. Wais'in `dizinleyicisi` olan `waisindex` komutu tek başına neleri/nasıl `dizinleyebileceğini` belirtir. Bu, `sürümden sürüme` ve `derlemeye` bağlı olarak bazı farklılıklar gösterebilir. `Sunucu` olarak `waisserver`, `istemci` olarak da `waissearch`, `waisq`, `swais`, `Xwais`, `Xwaisq`, `ws`, `xwais.sh`, `xwaisq.sh` sayılabilir. `Xwais client` programında, kullanıcı 'similar to' opsiyonu ile taramayı daraltabilir. Daha sonra kullanıcı, bu dokümanları işaret ederek kopyasını alabilir. `X` ve `vt100` istemci programları vardır.

Wais ailesinde başlıca ürünler ve şu anki sürümleri şunlardır: wais-8-b5.1, freeWais-0.5, freeWais-sf-2.xx, Isite-1.07.

wais-8-b* serisini bugünlerde kullanan az. freeWais'in gelişmesi durdu. CNIDR (Clearinghouse for Networked Information Discovery and Retrieval) merkezi (ftp://ftp.cnidr.org) Isite'i geliştiriyor. Bazı eklemelerle Dortmund Üniversitesi freeWais-sf'i (ftp://ls6-www.informatik.uni-dortmund.de), ve wais'e kolay bir arayüz (interface) olan Sfgate'i ve Waisperl'ü güncellemektedir.

A. Waisindex

Üzerinde tarama yapılması istenilen dosyaları, taramaya hazırlamak için 'waisindex' programı çalıştırılarak birçok yardımcı dosyadan oluşan bir dizin (directory) yaratılır. Bazı örneklerle açıklayacak olursak:

```
waisindex -d JARGON -t para jargon.txt
```

İçinde bulunduğumuz dizinde (directory) bulunan jargon.txt dosyasını, boş satırlarla ayrılmış birimleri birer doküman varsayarak dizinle, adı JARGON olsun ve bu dizinde (directory) dursun.

Bu işlemin sonunda bir JARGON.src dosyası ve bir sürü dizin dosyası yaratılacaktır. Bir .src dosyasında ilgili wais veri tabanının adı, kaynak olarak kullanılan dosyalar, düzenleyicisi, kısa bir tanımlanması ve tanıtıcı anahtar kelimelerin listesi vardır.

Bir diğer örnek:

```
waisindex -d /wais/INFO -t text -export -r /ftp/pub/INFO
```

Ftp'nin INFO dizini (directory) ve altındakileri dizinle, doküman adı olarak tüm patikayı al, veri tabanının adı INFO olsun ve wais dizininde (directory) bulunsun. export, veri tabanını bir wais sunucusu kanalıyla kullanıma açmak için bir hazırlıktır.

Böyle bir dizinde (directory) tüm wais veri tabanı dizin ve .src dosyalarının toplanması, tümünü waisserver kanalıyla ve wais://host/INFO benzeri URL ile sunmaya olanak sağlamaktadır.

Genelde dosya, doküman ve doküman başlığı farklı nesnelere. Her dokümanın bir başlığı vardır: Bu, tarama sonucunda istemci program kanalıyla kullanıcıya sunulur. Kullanıcı doküman başlığına bakarak dokümanı ister ve sunucu ilgili dokümanı gönderir. Bir dosya birden fazla doküman içerebilir. Bunu belirleyen -t ya da type opsiyonudur. Bazı önemli seçenekler:

text; her dosya tek bir doküman olarak alınır, tamamı dizinlenir, ve tüm patikası doküman başlığı olarak kullanılır;

filename; text'ten farklı doküman başlığı tüm patikayı değil sadece dosya adını içerir;

first_line; text'e benzer, başlık olarak ilk satır kullanılır;

one_line; dosyadaki her satır ayrı bir doküman olarak alınır, her satır aynı zamanda başlık olarak da kullanılır;

dash; dosya içinde en az 20 tire (-) içeren satırlar doküman ayracı olarak kullanılır, ve bir sonraki satır başlık olarak alınır.

Gif, tiff, pict türleri için sadece dosya adı dizinlenir ve başlık olarak kullanılır. TeX, dvi ve postscript dosyalarının içeriği dizinlenir, ve dosya adı başlık olarak kullanılır. mail'de ise her mesaj ayrı bir doküman olarak dizinlenir, ve 'Subject:' satırı doküman başlığı olarak kullanılır. Benzer bir biçimde her 'usenet' haberi ayrı bir doküman olarak alınır.

Birkaç örnek daha verelim. Diyelim ki /usr/ftp sizin anonim ftp dizinini (directory). Bunun altında yer alan INFO'yu dizinleyip URL olarak sunmak istiyorsunuz:

```
waisindex -d INFO -export -t URL /usr/ftp \
ftp://ftp.alfa.edu.tr -r /usr/ftp/INFO
```

Bunun sonucunda doküman başlığı onun ftp'ye göre URL'i olacaktır.

-M seçeneği, sizin aynı zamanda bir dosyanın text, gif, au, ps gibi birden fazla dokümanını kullanıcıya sunmak istediğinizde kullanabilirsiniz. Bunun için dosya adı uzantılarının TEXT, GIF, PS, AU gibi büyük harf olması, bir-biriyle ilintili dosyaların ilk adının aynı olması gerekir: baykus.TEXT, baykus.AU, baykus.GIF, baykus.PS, baykus.DVI gibi. Diyelim ki kuşlar dizininde (directory) bu tip dosyalar var:

```
waisindex -d birds -export -M AU,PS,DVI,GIF,TEXT \
/usr/ftp/pub/kuşlar
```

komutu TEXT dosyalarını dizinler. Bir kullanıcı baykus.TEXT'i seçince ona bu beş alternatif sunulur. Burada önemli olan TEXT türü dosyalardır, ötekileri ekleyip çıkartabilirsiniz.

-T seçeneği waisindex'in -t ile tanımadığı dosyalar içindir. -T mytype komutunu verirseniz wais sunucusu istemciye bunu 'mytype' türü olarak sunacaktır. Normal olarak bu dosyaların içeriğini dizinlemeyecektir ve başlık olarak dosya adını kullanacak ve onu dizinleyecektir. Böyle bir dosya seçildiğinde wais sunucusu 'binary' bir dosya olarak iletacaktır. Dosyayı ne yapacağı istemci programına kalmıştır. Standart MIME tipi için sistem temelindeki 'mailcap' ya da kullanıcının ~/.mailcap dosyasında bir işlem tanımlanmışsa o uygulanacaktır. Örnek olarak Web'in bir dizininde (directory) mevcut html dosyalarının içeriğini dizinleyip sunalım:

```
waisindex -d webtip -T HTML -export -contents \
/home/httpd/htdocs/tip/*.html
```

Bir Web uygulama programı (lynx, netscape, mosaic vb. ile sorgulama yaptığımızda sorgulama sonuçları size html dokümanları olarak sunulacak ve herhangi bir Web'den doğrudan alınmış etkisi yapacaktır.

B. Wais Dokümanlarının Sunulması

Wais dokümanları ya wais, ya gopher, ya da Web sunucusunun yerel makinede çalışan bazı programları kanalıyla sunulabilir. Cern, WN, GN gibi sunucular doğrudan wais'i desteklerler.

Ama Netscape, Explorer gibi yaygın sunucular wais URL'lerini desteklemezler. Lynx, şayet wais desteği ile derlendiyse, wais URL'lerine doğrudan erişebilir.

Doğrudan bir wais sunucusu çalıştırıp dokümanları sunma Akgül'ün çalışmasında anlatılmıştır (Akgül, 1996a).

Web üzerinden doküman sunma mekanizmalarından sadece wais.pl, wwwwais, SFgate ve gopher örneklerle anlatılmıştır.

FreeWAIS-sf'i geliştiren ekip wais veri tabanlarını Web'de sunmak için SFgate yazılımını geliştirdi. Perl 5.002'i isteyen SFgate, Perl'ün bir uzantısı olan waisperl'i ister. CPAN (Comprehensive Perl Archive Network) içinde Wais-2.xx.xx.tar.gz olarak bulunan waisperl'ü ftp.metu.edu.tr, ftp.bilkent.edu.tr ve akgul.bcc.bilkent.edu.tr'in ftp/http arşivinde bulabilirsiniz. FreeWAIS-sf dışındaki wais sürümleriyle de çalışan SFgate, freeWAIS-sf, waisperl ile çok daha hızlıdır. Yerel makinede bir wais sunucusuna gerek duymadan çalışabilir. Çeşitli örnekler <http://www.bilkent.edu.tr/SFgate/> adresinden görülebilir. Bunları derlemek için, ilgili dizinde perl Makefile.PL komutunu, daha sonra da make; make test; make install komutlarını vermek yeterlidir. Wais-perl kurulma aşamasında freeWAIS-sf'in derlenmiş olmasını, binary'leri, lib'leri ve 'include' dosyalarının yerini soracaktır.

i) Wais.pl

Sanal miting'in dokümanları /akgul/etc/httpd/htdocs/sanal/ dizininde (directory) bulunabilir. Bu dizinin URL'i: <http://www.bilkent.edu.tr/sanal/>. Bu dizindeki .html dosyalarını -t URL olarak dizinlemek için /akgul/etc/wais dizininde (directory) verilen:

```
waisindex -d sanal -t URL /akgul/etc/httpd/htdocs \
http://www.bilkent.edu.tr \
/akgul/etc/httpd/htdocs/sanal/*.html
```

komutunu girmek yeterlidir. NCSA (National Center for Supercomputer Applications) tarafından geliştirilen httpd yazılımıyla gelen cgi-bin'de wais.pl'i şu şekilde uyarlayabiliriz (sanal.pl adıyla):

```
#!/usr/local/bin/perl
$waisq = "/usr/local/bin/waisq";
$waisd = "/akgul/etc/wais";
$src = "sanal";
$title = "Sanal Miting .html files ";
```

Dosyanın geri kalanında değişiklik yapmaya gerek yok. <http://www.bilkent.edu.tr/cgi-bin/sanal.pl> URL'i ile ilgili veritabanını sorgulayabiliriz. Wais.pl'in gelişimi olarak "Son of Wais.pl" ve "kidowais.pl" ortaya çıkmıştır. Yahoo'da bunlara ait bağlantı bilgisi bulunabilir.

ii) SFgate ile Sunum

Biraz daha farklı dinleyerek her mesajı ayrı bir doküman olarak sunmak için, mesajlar arasında boş satır bırakılır. Bir mesajın içinde boş satır olması gerekir. Bu durumda

```
waisindex -d sanal2 -contents http://www.bilkent.edu.tr\
/akgul/etc/httpd/htdocs/sanal/*.html
```

komutuyla dinlenen sanal2, wais.pl ile sunulmazken, aşağıda belirtildiği gibi SFgate ile gayet düzgün sunulabilmektedir.

```
<INPUT NAME="database" TYPE="checkbox" VALUE="local/sa-
nal" >
Bilkent WAIS server: Sanal Miting Dokumanlari <BR>
<INPUT NAME="database" TYPE="checkbox" VALUE="lo-
cal/sanal2" >
Bilkent WAIS server: Sanal Miting Dokumanlari - II <BR>
```

Yukarıdaki örnekleri <http://www.bilkent.edu.tr/SFgate/bilkent-sf.html> içinde bulabilirsiniz.

Bir ÖSYM Denemesi: ÖSYM'nin sunduğu biçimde, sınav sonuçlarını wais ile sunmanın basit bir yolu şudur: Her öğrenciye ait kayıtları, bir dosya olarak

tutmak ve dosya adı olarak öğrencinin numarasını kullanmak ve dinlemek için:

```
waisindex -d osym -t filename -nocontents -r \
  /akgul/etc/httpd/htdocs/osym
```

komutu yeterli olacaktır. Sadece dosya isimleri dinlenmiştir. Dolayısıyla sadece numaralar kanalıyla erişim mümkündür. wais.pl bu bilgileri bu şekilde sunamazken, SFgate gayet güzel sunabilir.

iii) wwwwais ile Sunum

wwwwais.c programı wais'in Web'den sunulması için bir geçit görevi yapar. Kevin Hughes tarafından yazılan bu programın son sürümü wwwwais.tar.gz olarak Bilkent arşivinden ve <http://www.eit.com/software/wwwwais/> adresinden alınabilir. wwwwais.24.c'den itibaren konfigürasyon wwwwais.conf dosyasına kaydırılmıştır. Kaynak dosyasında konfigürasyonun yeri şu komutla belirtilir:

```
define CONFFILE "/usr/local/etc/httpd/conf/wwwwais.conf".
```

Derlenen program, /cgi-bin/'e yerleştirdikten sonra <http://host/cgi-bin/wwwwais> URL'i kullanıcıya wwwwais.conf'da tanımlı veri tabanlarını bir 'pop-up' mönü olarak sorgulama için gerekli tarama penceresiyle birlikte sunar. Bu program waisq, waissearch veya swish'i arama motoru olarak kullanır. Waissearch için, wais sunucusunun çalışıyor olması gerekir. Dokümantasyon yukarıda verilen dosya ve URL'de bulunabilir.

iv) Gopher'dan Wais Sunulması

Wais desteğiyle derlenen gopher sunucusu ile: i) Kendi gopher dizinlerimizde (directory) kuracağımız wais veri tabanlarını sunabiliriz, ii) Dünya üzerindeki diğer wais sunucularına erişim olanağı sağlayabiliriz. Doğal olarak Web tarayıcılarıyla da yukarıdakilere erişebilirsiniz.

Bu durumda Gopher Data dizininde (directory) herhangi bir veri tabanı dizinlendiğinde, onun oluşturduğu .src dosyası gopherd tarafından algılanıp sunulur. Kullanılan kural, bir .index dizini yaratıp içindeki veri tabanını da index olarak adlandırmaktır. Gopher paketinin içinde 'gopherindex'

shell script'i bu amaçla kullanılabilir. gopherd 'man' (manual) sayfalarında da örnek vardır. Esas olarak bunun herhangi bir wais veri tabanı oluş-

turmaktan bir farkı yoktur. Wais dizinini `.index` dizininde `index.src` adıyla yaratmak gereklidir.

Isite, Zserver

CNIDR ekibi Z39.50'nin ikinci sürümünü temel alan yeni bir yazılım geliştirmektedir. **Isite** olarak bilinen bu yazılımın parçaları "zserver" (waisserver'e benzer), "sapi" (Search Application Programming Interface), "Iindex" dizinleyici ve "Isearch" tarama programıdır. Iindex ve Isearch, Isearch-xxx.tar.gz olarak ayrıca alınabilir. Zserver'da ayrıca paket olarak bulunabilir. Isearch'in Web'de kullanılabilmesi için Isearch-cgi paketi de vardır.

Bu paketleri `ftp://ftp.cnidr.org`'dan ya da `ftp://ftp.bilkent.edu.tr`'dan alabilirsiniz. Derlemek için `c++` ya da tercihan GNU'nun `g++`'inin kullanılması gerekir. En son `g++` ve `libg++` (ya da bir önceki) programları ile kolayca derlenebilir. 'make' komutu derlemek için yeterlidir.

Bu konuda henüz dokümantasyon az. Isearch ve Iindex programları, hiç argüman verilmeyince, kullanımı ekrana yazarlar. Opsiyonlar waisindex'inine benziyor. Henüz yeni olduğu için `-t` ile desteklenen türler az, ama `-s` opsiyonu yeni; bu opsiyon dokümanları ayıracak ifadeyi belirtir. Wais'e kıyasla daha küçük dizin oluşturuyor.

Isearch programı waissearch'e benzer, ama algoritmada bazı farklılıklar var. Isearch, waissearch'e göre daha yetenekli. Şayet veri tabanındaki kayıtlar belirli alanlara ayrıldıysa alanlar üzerinde tarama yapabilirsiniz; "beaut*" de olduğu gibi örnek taraması yapabilirsiniz, bazı kelimelere daha fazla ağırlık verebilirsiniz.

Iindex ve Isearch'i Web içinden kullanmak için Isearch-cgi paketi vardır. "isrch_fetch", "isrch_form", "isrch_srch" binary dosyaları ve "ifetch", "iform", "isearch" script'leri vardır. Script'leri gözden geçirip, Iindex dizinlerinin ve binary dosyaların yerini belirtip, cgi-bin'e yerleştirip, tarama için bir html dosyası koymak yetmektedir.

Isite paketinin diğer parçaları; Iindex dosyalarını temizlemek için bir "Iutil", sunucu "zserver", ve istemci "zclient"dır. Bunların çalışması için "sapi.ini", "zserver.ini", "zclient.ini" dosyalarının uyarlanması ve yerleştirilmesi gerekir. `zserver i, /path-to-zserver.ini` şeklinde bir komut (standalone) ya da ona eşdeğer komutla `inetd`'den çalıştırmak mümkündür. zserver bazen "Zdist" olarak da bilinir. Z39.50 sunucusu, istemcisi dışında bir http geçit makinesi olarak da kullanılabilir. Maalesef, henüz yeterli dokümantasyon yoktur.

Isite'in bir parçası olan `zgate` ile gittikçe artan bir oranda dünyadaki kütüphaneleri Z39.50 ile sorgulayabilirsiniz. Burada farklı olan, Web üzerinden kullanıcıdan sorgulama verilerini alıp, onu Z39.50'ya aktaran, ortaya

çıkan sorguyu ilgili Z39.50 sunucuya gönderen ve gelen cevabı, html'e aktaran yazılım yerel makinede çalışmakta, bu ise kıt olan hat kapasitesini daha etkin kullanmakta, geçit makinalarını ortadan kaldırarak daha hızlı bir servis olanağı sağlamaktadır. ABD Kongre Kütüphanesinin Web'inde zgate çalışmaktadır.

Glimpse

Glimpse (GLobal IMPLICIT Search) agrep'i (approximate grep) temel alan bir dosya tarama programıdır. Taramada bir miktar hatayı kabul eden, dizini daha az yer tutan bir programdır. Glimpse agrep'de kabul edilebilecek hata sayısını, dönecek dokümanların ayracını belirtebilir, sadece tam kelimeleri bulmasını, küçük harf büyük harf farkını unutmamasını söyleyebilirsiniz.

Şu anda glimpse-4.0 ve glimpseHTTP2.0 var. ftp://ftp.cs.arizona.edu/glimpse/ , ftp.bilkent.edu.tr, ve ftp://ftp.metu.edu.tr'dan kaynaklar alınabilir. http://glimpse.cs.arizona.edu/ adresinde ise 'Glimpse Home Page'ı bulunur.

Hemen tüm işletim sistemleri için hazır 'binary'leri ve kaynaklar içinde işletim sistemi için 'Makefile' dosyaları bulunmaktadır. Mekanizma wais gibidir: glimpseindex ile dizinlenir; şayet dışarı servis verilmek isteniyorsa, glimpseserver çalıştırılır, taramalar ise glimpse ile yapılır.

Glimpse'i Web içinden etkin kullanmak için, GlimpseHTTP, GlimpseGate ve Webglimpse yazılımları var. GlimpseHTTP ve GlimpseGate örnekleri, Bilkent Web'inde "Internet Services in Turkey" ve başka yerlerde bulunabilir.

Glimpse Dizinlerinin Web'den Sunulması

GlimpseHTTP'yi kurmak için: yeni bir dizin (directory) yaratıp, içinde glimpseHTTP-2.0-src.tar.gz paketini açın, **ghinstall**'i çalıştırın. Bazı programlar için bir dizin (directory) seçmek ve Web'in kuruluşu hakkında bilgi vermek zorundasınız. Script'ler ve bazı destek dosyalar için, /usr/local/bin gibi her kullanıcının erişebildiği bir dizin seçilmesini öneririz.

Daha sonra esas olarak dizinlemek istediğiniz bir dizinde (directory) **makegharc** script'ini çalıştırır, bazı sorulara cevap verirsiniz; glimpseindex ghindex.html dosyasını yaratır. glimpse_exclude, .glimpse_include dosyaları kanalıyla nelerin dizinleneceğini denetlersiniz. Glimpse, normal olarak 'soft link'leri izlemez; bu tür dosyalar dizinlenmez. Bu tür dosyaları **.glimpse_include** içinde belirterek dizinlenmelerini sağlarsınız. ghindex dosyası

kanalıyla yapılan taramalarda ilgili dokümanlarda aranan kelimenin geçtiği satır gösterilir. GlimpseGate ise, o kelimenin içinde bulunduğu paragrafı kullanıcıya tarama sonucunda hemen sunar. Pek çok halde kullanıcı zaten o paragrafı görmek istemektedir. GlimpseGate, buna karşılık daha büyük bir izin kullanabilir; kurulması zor olmamakla birlikte daha uzundur. Yani daha fazla Perl modülü kurulması gerekmektedir. Bunların listesi GlimpseGate dokümantasyonunda vardır: libwww-perl-5xxx.tar.gz, CGI.pm-2.xx.tar.gz, libnet-1.00.tar.gz, Search-0.2.tar.gz, Socket-1.5.tar.gz. Perl5 modüllerinin kurulması perl Makefile.PL komutundan sonra make; make test; make install komutlarıyla olmaktadır. SFgate ve perl-wais'de de aynı yöntem uygulanmaktadır. GlimpseGate kurulduktan (perl modüllerinin kurulması, Glimpsgate ve onun 'soft link' kardeşi nph-Glimpsegate cgi-bin'e yerleştirildikten) sonra .glimpsgate_conf dosyası glimpseHTTP kurulmuş bir dizine (directory) yerleştirilmiş ve uyarlanmış olmalıdır. Bu durumda

```
/cgi-bin/nph-GlimpseGate?DIR=/akgul/etc/httpd/htdocs/sanal
"Sanal miting" dizininde GlimpseGate taramasının URL'i olmaktadır. Burada DIR='in Web'e göre değil, gerçek patikayı belirtmesine dikkat edilmelidir.
```

Webglimpse ise glimpseHTTP'den sonra Glimpse ekibince geliştirilmiş daha yetenekli bir programdır. Verilen bir URL içindeki bağlantıları iki adım takip edip, o dokümanları dizinleyebilir, ya da bir dizindeki (directory) ve altındaki dokümanları dizinleyebilir. Bir Web'i tüm olarak dizinleyip, dizin temelinde ve tüm Web üzerinden tarama olanağı sunabilir. Kurulması ve çalıştırılması glimpseHTTP'ye benzer.

Swish

Swish oldukça yetenekli dizinleme amaçlı bir programdır. Swish dizini wais'e göre daha az yer tutar. Bir uygulaması Unix Help sayfalarındadır. Bunun bir kopyasını, Bilkent'te <http://www.bilkent.edu.tr/Online/unixhelp> veya http://www.bilkent.edu.tr/cgi-bin/unixhelp_search URL'leriyle görebilirsiniz. "swish.conf" gibi bir konfigürasyon dosyası ile opsiyonları belirlersiniz. Bir dizini (directory) ve altındaki dosyaları dizinleyebilirsiniz.

Wais'deki **-t URL**'den daha gelişmiş bir şekilde kullanıcıya sunulacak dosya adını belirlemenize olanak verir. Kendi özel dosyalarınızı dizinleyip, tarama için kullanabilirsiniz. Web'den sunmanın bir örneği, Unix Help dosyaları için kullanılan perl script'idir. Diğer bir yol ise wwwwais programıdır; bu program swish'le dizinlenen dokümanların Web'den sunulmasına izin verir. Bilkent'te <http://www.bilkent.edu.tr/WWW/Swish/> altında kaynak ve dokümantasyon bulunabilir.

Diğerleri

Bu makalede çok yaygın olarak kullanılan bazı tarama mekanizmaları üzerinde durulmuştur. Bunlar dışında kalan diğer bazı tarama mekanizmalarından da kısaca bahsetmekte yarar var. **ffw** (Freetext search For Web) wais'den çok daha az yer tutan, html kurallarını anlayan ve daha etkin bir tarama programıdır. Boole işlemleri ile tarama yapmayı destekler ve birden fazla dizini tarayabilir. Kullandığı dizinler, kendi başına sonuçları kullanıcıya sunmak için yeterlidir; dizin ve ona kaynak olan dokümanlar başka makinelerde tutulabilir. Kaynağın esas merkezi <http://www.nta.no>'dan alınabilir. Bilkent'te de kopyası vardır; Yahoo'dan izi bulunabilir.

Gloss Stanford, **Harvest** ise Colorado Üniversitelerinde geliştirilmiş mevcut dizinlerden yeni dizinler üretmeye yönelik programlarıdır. Bu programları kendi Web'inizde yer alan dokümanları dizinlemek için de kullanabilirsiniz. NCSA'de geliştirilen wais temelli **docfinder** paketi bir Web'deki dokümanları tarama mekanizmasıyla sunmaya uygundur. **Excite** ticari bir firma tarafından geliştirilen, ama 'binary' olarak parasız dağıtılan etkin ve yaygın bir tarama mekanizmasıdır.

Bir veri tabanı motoruyla etkileşime girip, kullanıcıya Web üzerinden veri tabanına erişim olanağı sunan pek çok geçit programı vardır. Bunlarla ilişkili bağlantı bilgileri Yahoo'da bulunabilir.

Washington Üniversitesinde geliştirilen ve birkaç yıldır kullanımda olan, 94 ortasından beri dağıtımı yapılan bir paketten de bahsetmekte yarar var. **Willow** bir X/Motif uygulama programı olup genel amaçlı enformasyon erişimi için tasarlanmış bir programdır. Homojen olmayan pek çok veri tabanına erişilmesini, kolayca enformasyon taraması yapılmasını ve sonuçların alınmasını sağlıyor. **Wilco**, Willow'un vt100 ekranında çalışan kardeş programıdır. Wilco'nun MS Windows için de bir sürümü hazırlanmaktadır. Willow, Washington Üniversitesinde 250 küsür X-terminalde kullanılmaktadır. Dünya üzerinde 1200 kurum tarafından da en azından denenmiştir.

Willow projesinin en ilginç yönü, bu yazılımın çeşitli veri tabanlarıyla **database-driver** programları kanalıyla konuşmasıdır. Bu konuşma bir unix 'pipe' üzerinden mesajların iletilmesi şeklinde olmaktadır. Şu anda Willow, Washington Üniversitesinde BRS ticari yazılımının sunucusu ile telnet üzerinden, Z39.50 sunucusu ile TCP/IP üzerinden kullanılabilir.

Sonuç

Bu makalede Internet aracılığıyla erişilebilen elektronik bilgi kaynaklarını dizinlemek ve herkesin kullanımına açmak için yaygınlıkla kullanılan bazı