Peter F. Lemkin[1]
Michael Chipperfield[2]
Carl Merril[3]
Steven Zullo[3]

[1]Image Processing Section/LMMB,
NCI-FCRDC/NIH, Frederick,
MD, USA
[2]GDB, John Hopkins University,
Baltimore, MD, USA
[3]LBG/NIMH, Washington DC, USA

A World Wide Web (WWW) server database engine for an organelle database, MitoDat

We describe a simple database search engine "dbEngine" which may be used to quickly create a searchable database on a World Wide Web (WWW) server. Data may be prepared from spreadsheet programs (such as Excel, *etc.*) or from tables exported from relational database systems. This Common Gateway Interface (CGI-BIN) program is used with a WWW server such as available commercially, or from National Center for Supercomputer Algorithms (NCSA) or CERN. Its capabilities include: (i) searching records by combinations of terms connected with ANDs or ORs; (ii) returning search results as hypertext links to other WWW database servers; (iii) mapping lists of literature reference identifiers to the full references; (iv) creating bidirectional hypertext links between pictures and the database. DbEngine has been used to support the MitoDat database (Mendelian and non-Mendelian inheritance associated with the Mitochondrion) on the WWW.

1 Introduction

The World Wide Web (WWW) [1] is a vast set of inter-connected documents available over the Internet that has changed the way scientists communicate their data. It offers the opportunity to publish data in a way that is immediately accessible to the scientific community who have access to the Internet. The dbEngine program de-scibed in this paper is a simple database search engine used with a WWW server. It organizes and allows easy access to data that a Web server provider wants to make available. The user interacts with the WWW using a Web browser (*e.g.* Mosaic, Netscape, *etc.*). This may be thought of as a client. The WWW server that responds to client requests supplies data to the client when requested – it serves the client. We will be describing dbEngine, which is that part of the WWW server that handles particular data requests directed towards it.

Web servers are available commercially as well as being publicly available from NCSA (over the Internet at ftp.ncsa.uiuc.edu). Because it is simple to set up a Web server, the ability to export local data prepared from spreadsheet programs (such as Excel) or from tables exported from relational database systems makes this means of communication and interaction attractive to organizations wishing to make their data available.

We will describe the dbEngine capabilities listed in the abstract and then show some examples. The database

engine has been used with several databases to provide WWW access. The one used here to describe dbEngine is MitoDat, Mendelian Inheritance and the Mitochon-drion (S. Zullo *et al.*, in preparation; URL http://www-lmmb.ncifcrf.gov/mitoDat/; its "home page" is shown in Fig. 1). Another is the *Annual Reviews* index (not dis-cussed here) with (URL http://www-lmmb.ncifcrf.gov/annRev.html). We assume that the reader has some knowledge of the WWW to fully understand the details of dbEngine we present. We hope our description will help an occasional Web user to see the role such a data-base engine may play, even without understanding the full details of how this happens.

1.1 The MitoDat database

MitoDat is primarily dedicated to human nuclear genes specifying the enzymes, structural proteins, and other proteins — many still not identified — involved in mito-chondrial biogenesis and function. MitoDat predomi-nantly highlights human mitochondrial proteins, al-though we are including proteins from other animals, yeast, fungi, and plants. During evolution, mitochon-drion import systems have developed to gather metabo-lites from the surrounding cytoplasm. The mitochon-drion realized a drastic reduction in the coding capacity of its genome through the evolution of import systems gathering proteins from the surrounding cytoplasm, with consequent or concurrent mutational inactivation or loss of functional genes. As a consequence, the mitochon-drial genome evolved to be small.

This database consolidates information from the various biological databases, *e.g.* GenBank, SwissProt, Genome Data Base (GDB), Online Mendelian Inheritance in Man (OMIM), Mouse Genome Database, Protein Disease Database (PDD) [2, 3]. Some of the concepts used in dbEngine are similar to those used in the PDD. Because the mitochondrion has a central role in cellular metabo-lism, it is involved in many human diseases, so the MitoDat database should help us in studying these diseases.

Figure 1. Home page for the MitoDat database. From the home page, one may search, download the entire MitoDat database and references, and access the interactive chromosome idiograms.

In most animals the mitochondrial genome now encodes only 13 proteins of the thousands necessary for its biogenesis and functions. All 13 are inner membrane proteins and involved in the electron transport chain or ATPase [4]. The use of two-dimensional gel electrophoresis for the identification of mitochondrial proteins was pioneered by Anderson [5] who labeled cells with [^{35}S]methionine in the presence or absence of mitochondrial inhibitors. Thus nuclear-encoded mitochondrial proteins are prevented from being imported into the mitochondria [5, 6]. As a consequence, tissue-culture cells labeled with [^{35}S]methionine in methionine-free medium in the presence of 100 µg/mL emetine, an inhibitor of cytoplasmic protein synthesis, may be used to identify mitochondrially encoded proteins [7]. Chromosome idiograms labeled with the mitochondrial genes for the human genome are available as part of MitoDat. By clicking on a gene name in a idiogram in the Web browser, the corresponding database protein entry is displayed. The user may also search for proteins by keyword.

2 Materials and methods

The dbEngine is a Common Gateway Interface (CGI-BIN)* program written in C. It runs on SUN or other UNIX workstations running the NCSA "httpd" WWW server.

---

* McCool, R., 1994, NCSA, University of Illinois at Urbana-Champaign, http://hoohoo.ncsa.uiuc.edu/egi, 1994
NCSA, 1994, Mosaic for X Version 2.0 Fill-Out Form Support. NCSA, University of Illinois at Urbana-Champaign, http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/Docs/fill-out-forms/overview, html

Its role is to process the search query that is passed to it from the user's Web browser program.

2.1 Use of dbEngine for searching a WWW database

This general-purpose WWW database engine may be used to answer queries from any simple sequential table database exported from various spreadsheets (such as Excel, *etc.*) or single tables exported from relational database systems (such as Oracle, Sybase, dBaseIV, *etc.*). Because of these minimal constraints and its easy configuration, dbEngine is able to use a wide variety of easily exported data in constructing Web server databases. The data collection method for MitoDat and its preparation using an Excel spreadsheet is described by Zullo *et al.* (in preparation).

2.2 Description of the processing methods

When a search is performed, the name of the database is specified in the Web browser search criteria. This is used by dbEngine to look up the actual database file, which is then read one record at a time. All database files are ASCII text data files consisting of records — one record per line. The format is simple enough so that data may be prepared with a text editor. A database record consists of a row of delimited field (*i.e.*, spreadsheet column) values. Typically the delimiter is a tab or comma. DbEngine uses a simple search mechanism, a strict conjunction (AND) or disjunction (OR) of search terms that are applied to each record independently. Each term of the search expression is then tested against the entire row and dbEngine notes whether the term matches the record. Then the associated Boolean AND/OR connectives of the search expression are evaluated and the row is accepted or rejected as a whole; for example, sdh AND Inner is the search expression we used in the example shown in Section 3.

Each database row that matches the search criteria is then translated into hyper text markup language (HTML) used with hyper text transfer protocol (HTTP) [1]. Finally, the HTML that was generated is returned to the Web browser that invoked the search. This appears to the user as a page of text containing the search results. The following shows the generic format of the HTML generated for the fields and their associated values.

```
<DT> <B>field-1</B> = value-1
<DD> <B>field-2</B> = value-2
        . . .
<DD> <B>field-n</B> = value-n
```

This results in the following display to the user:

field-1 = value-1

   field-2 = value-2

   ...

   field-n = value-n

Note that fields 2 to *n* are indented. A specific example is shown in Figs. 4 and 5 in Section 3.

2.2.1 Reporting values as hypertext links

The database configuration can designate particular fields whose data is to be returned as hypertext links pointing to other biomedical database Web servers. This makes it easy to automatically generate hypertext references to other biomedical Web servers that specialize in particular types of data and share them over the Internet. This assumes that the value is a) an outside WWW server database identifier and b) it is appended to a base value uniform resource locator (URL) to synthesize a URL to that database. URLs are used to specify hypertext documents in the World Wide Web [1]. To illustrate this, we give an example where we denote the base URL as base-URL. Then, for entry j, the *value*$_j$ is translated to a hypertext link. In the following example, the URL is shown relative to the Web server base-URL for *value*$_j$.

```
<A HREF="/cgi-bin/dbEngine?mitoDat,getDataByID,valuej">valuej</A>
```

where the base-URL is:

```
/cgi-bin/dbEngine?mitoDat,getDataByID,
```

2.2.2 Mapping lists of "reference" identifiers to full references

Finally, a particular field may be designated as the "reference" field consisting of a set of reference identifiers (delimited by ";"s). By references we mean, for example, literature references. After a search, if reference data exist, these identifiers are mapped to the full references through a local reference mapping file. If the reference identifiers are to be mapped to hypertext, then this is done as well. For example, if the reference identifier list is:

```
Aas68; Alessio67; Avigan61
```

this is translated to the hypertext:

```
<UL>
<LI> [<B>Aas68</B>] Aas and Bremer 1968 Biochim. biophys. Acta 164 157-166
<LI> [<B>Alessio67</B>] Alessio et al. 1967 Ital. J. Biochem. 16 161-166
<LI> [<B>Avigan61</B>] Avigan and Steinberg 1961 J. biol. Chem. 236 2898-2900
</UL>
```

2.2.3 Database entries and objects in images

MitoDat connects labeled genes in chromosome idiograms to MitoDat entries, and conversely from MitoDat database entries to the corresponding chromosome idiogram(s). The image-to-database correspondence is done using the HTML ISMAP [1] option shown for chromosome 1 using the dbEngine ismap function — where hum_01 is the name of the GIF image for the idiogram. Figure 2 shows the labeled idiogram for chromosome 1. This hypertext causes an image to be displayed.

```
<A HREF="/cgi-bin/dbEngine/mitoDat,ismap,hum_01">
<A IMG SRC="hum_01.gif"> </A>
```

Clicking on this image in the Web browser activates a special hypertext link. In our case, clicking on a chromosome label appends the *x,y* coordinates in the image to the /cgi-bin/dbEngine/mitoDat,ismap, *image-name as* ?*x,y* path-information. The ISMAP image coordinates are specified in "pixel raster" form with (0,0) being the upper left-hand coordinate. The *image-name*?*x,y* data is then
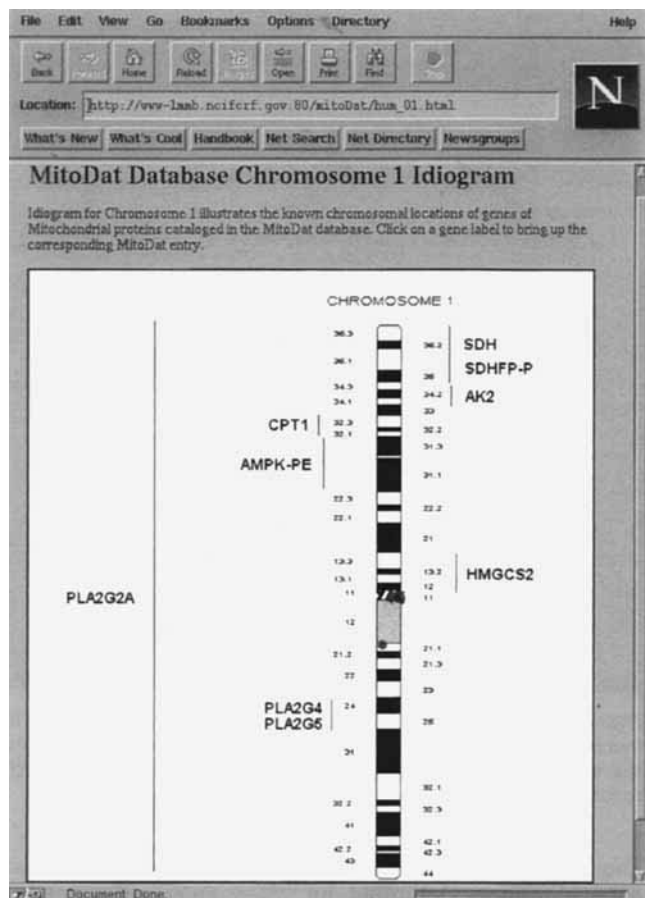
Figure 2. Labeled chromosome idiogram for the MitoDat database. By using the dbEngine ismap function, clicking on a label (*i.e.*, gene) invokes the database entry for that gene. The idiograms were derived for the "Idiogram Album: Human" © 1994 by David Adler.

used to search a mapping file to look up a corresponding MitoDat database entry identifier. It then invokes the search as if the user had typed in the identifier directly to do a search. Mapping the other way from a database hypertext entry to an image is simpler. Images are embedded in HTML documents as shown in the above HTML code desciding the ismap function. Then a database entry field (*e.g.* idiograms) is simply translated to the HTML document containing the chromosome idiogram image.

2.3 Data file preparation — using a spreadsheet program

The dbEngine uses simple sequential data files that are easily maintained in, and exported from, spreadsheet programs or text editors. Thus it is not difficult to extend or modify a database with new fields using these commonly available programs. It is also easy to have the dbEngine generate hypertext links to them.

2.4 Description of the database files

All the database files are ASCII text files and have the same database name but different file extensions. This allows several different databases to be kept in the same directory (*e.g.*, we keep them in our anonymous file transfer protocol, FTP, server pub directory). We will use

the name *dbName* here for purposes of discussion. The database name is supplied to the dbEngine when it receives a request for some action from the Web browser. Table 1 lists the files used in the dbEngine.

Table 1. Files used in dbEngine[a]

`dbEngine.conf`	– map database name to full path for server files
`<dbname>.f2u`	– database mapping file maps fields to URL base addresses
`<dbname>.txt`	– database table
`<dbname>.ref`	– references table associated with database table
`<dbname>.map`	– ISMAP database for images, regions and objects
`<dbname>.hdr`	– HTML header reported when a search is done
`<dbname>.rhd`	– HTML header reported when a references are reported
`<dbname>.tdd`	– optional data dictionary lines for database table
`<dbname>.rdd`	– optional data dictionary lines for references table

a) The basename of the data files is <dbName>-the name of a specific database. For the MitoDat database, it is mitoDat. This means that several databases may reside in the same directory if desired. The dbEngine.conf file maps the database <dbName> to the path where the corresponding files reside.

2.4.1 Configuration file dbEngine.conf

A database name configuration file dbEngine. conf maps the *dbName* to a specific base path on the Web server file system. This base path specifies where all the database files are kept for the dbEngine. For example, the two paths for MitoDat and the Annual Reviews databases might be:

```
mitoDat=/home/www-data/mitoDat
annrev=/home/www-data/annrev
```

2.4.2 Database table format

A database record is a list of delimited fields contained on one line. The delimiter is specified in the *dbName*.f2u mapping file and is generally either a tab or comma, for example, the record delimiter "," as:

```
@DB-delim=,
```

Data fields may be enclosed in quotes ("...") that are removed when the data is read. Database data is a single text file *dbName*.txt with two header records (lines) followed by any number of data records (lines). The database header contains: a) the revision date record, b) a data dictionary record listing the database field names for data that follows (a data dictionary gives the names of the fields for the data values in a database). For example, the following shows the two header records for the MitoDat database (here indented-lines denote continuation).

```
MitoDat database revised 29 June 1995
MitoDat ID Number, EC #, Protein Name, Organelle Compartment,
    Pathways/Enzyme Complex, Gene Symbol, Gene Name, GDB Accession Number,
    MIM Accession Number, PDD ID number, Cytogenetic Position,
    Mouse Gene Symbol, Mouse Cytogenetic Position, Idiograms,
    MGD Accession Number, Mouse Link, Drosophila Gene Symbol,
    Drosophila Cytogenetic Position, Fly Unique Identifier,
    Maize Gene Symbol, Maize cytogenetic Position, Maize Accession Number,
    Notes, References
```

Alternatively, both lines of the header could be put in a table data dictionary file *dbName*.tdd in which case the *dbName*.txt just contains data records.

2.4.3 Generating hypertext links

The correspondence between files and hypertext links is given in the *dbName*.f2u file. This 'field' to 'URL' mapping file specifies special fields for which a hypertext reference is generated in the report *if data exists* for that field. For example, the following shows some of the field to URL mapping directives used in the MitoDat database using the "=" and "=@" directives.

```
EC #=http://expasy.hcuge.ch/cgi-bin/get-enzyme-entry?
GDB Accession Number=http://gdbwww.gdb.org/gdb/gdbx?o=1&id=
MIM Accession Number=http://gdbwww.gdb.org/omim/omimx?
CitDb Accession Number=http://gdbwww.gdb.org/gdb/gdbx?o=9&id=
Mouse Link=http://www.informatics.jax.org/bin/fetch_locus?
Fly UniqueIdentifier=http://morgan.harvard.edu/htbin-post/gene.script?
Maize Accession Number=http://www.agron.missouri.edu/cgi-bin/sybgw/new/Locus/
PDD ID number=http://www-pdd.ncifcrf.gov/PDD-bin/gGetPDCbyID?pdc_id,tableForm,
Idiograms=@http://www-lmmb.ncifcrf.gov/mitoDat/
```

If the "=@" directive is used (*e.g.*, with the Idiograms field above), then the default ".html" hypertext document file extension is appended to the end of the translated *hypertext reference* after the identifier. The feature is useful if there is no search engine on the relevant Web database server, but just a set of hypertext HTML documents. We may then access these HTML documents directly, but must specify a .html file extension to do so. For example, the hum_01 idiogram identifier gets mapped to an http://.../hum_01.html HTML file name. Adding identifiers that translate to hypertext links is as simple as adding the new field to the database table with Excel and then adding the "field to base-URL" in the dbEngine mapping file.

2.4.4 Mapping image objects to database queries

The translating of rectangular regions in image to dbEngine database entries is done in the *dbName*.map file. The *x,y* data from the Web browser is matched against the coordinate ranges specifying a rectangular region for that GIF image. If a match is found, dbEngine uses the corresponding ObjectID value in a search and returns that search's results. For example, the following is part of the .map file for the MitoDat database (where # indicates a comment line),

```
# GifFilename ObjectName ObjectID x1 y1 x2 y2
hum_01 SDH MD300020 414 62 449 76
hum_01 SDHFP-P MD300025 417 87 482 100

   . . .
hum_02 CAD MD100091 95 107 126 121
hum_02 HK2 MD300066 275 190 308 203
```

2.4.5 Expanding reference identifiers to full references

Full literature references are kept in a *dbName*.ref file that is used for mapping reference identifier names to the actual references. The latter is optionally invoked if the DB.expandReferences "checkbox" is used (see example in Fig. 3). As with the *dbName*.txt file, the record header may be specified in a separate reference data dictionary file *dbName*.rdd in which case the *dbName*.ref contains just data records. For example, the References field data might be Aas68; Alessio67, which translates to:

- • [Aas68] Aas and Bremer 1968 Biochim. biophys. Acta 164 157-166
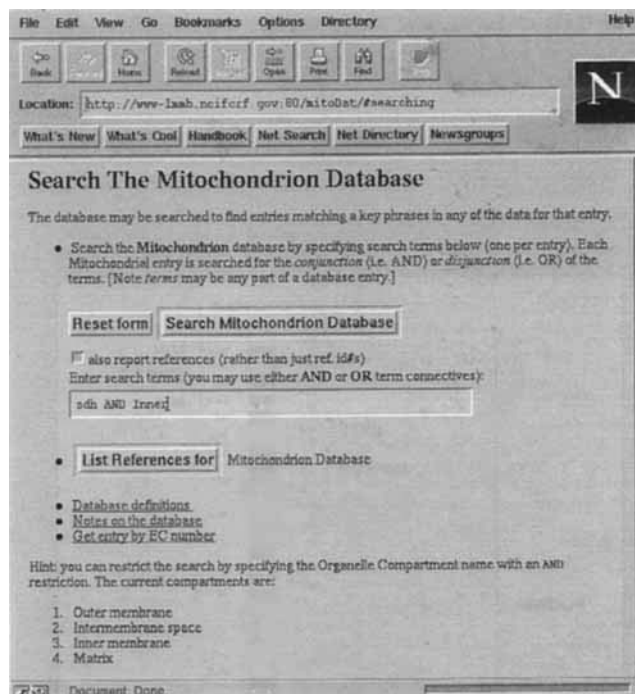- • [Alessio67] Alessio et al. 1967 Ital. J. Biochem. 16 161-166

Figure 3. Common Gateway Interface (CGI), search form used to frame a query. The Boolean search criteria are entered in the form and submitted to the Web server running the dbEngine. The search condition used was: sdh AND Inner.

The "References" field data may contain a list of reference identifiers delimited by ";" in *dbName*.txt entries. For the above MitoDat example, the references field and delimiter are defined in the *dbName*.f2u mapping file as:

```
@RefMapFieldName=References
@RefMapFieldDelim=;
```

There are two types of reports: one for a *search* and the other to generate a formatted *list of all references*. There are corresponding HTML report prefix files associated with these reports: *dbName*.hdr and *dbName*.rhd, used for search and reference reports, respectively. These are not shown here because of space limitations. These let the database maintainer incorporate other descriptions and hypertext in the resulting reports returned to the user.

2.5 Methods of invoking the search engine

We illustrate the four methods of invoking the dbEngine by showing the HTML code used on the Web browser to make requests of the database engine. Then we show some screens of a search. When the dbEngine is invoked by the Web server, it determines which of four functions (search, references, getDataByID, ismap) is requested and then gets its arguments from the Web browser accordingly. It then searches the database and returns the results to the Web browser.

2.5.1 Search using text form input – search

The first example illustrates how dbEngine is invoked from HTML to do a search on the MitoDat database.

This appears to a user as a portion of Fig. 3 where the CGI search form is used to frame a query.

```
<FORM METHOD="POST" ACTION="/cgi-bin/dbEngine?mitoDat,search">
<INPUT TYPE="reset" VALUE="Reset form">
<INPUT TYPE="submit" VALUE="Search Mitochondrion Database"> <BR>
<INPUT TYPE="checkbox" NAME="DB.expandReferences" VALUE="on" CHECKED>
also report references (rather than just ref. #s)<BR>
Enter search terms (you may use either <B>AND</B> or <B>OR</B>
term connectives):<BR>
<INPUT NAME="DB.grep" SIZE=55>
</FORM>
```

2.5.2 Get a list of references used in the database — references

The second example illustrates how dbEngine is invoked from HTML to return a list of references. This appears to a user as a portion of Fig. 3.

```
<FORM METHOD="POST" ACTION="/cgi-bin/dbEngine?mitoDat,references">
<INPUT TYPE="submit" VALUE="List References for"> Mitochondrion Database <BR>
</FORM>
```

2.5.3 Get a specific database entry using a unique identifier — getDataByID

The third example illustrates how dbEngine is invoked from HTML using a URL reference. It returns specific entries indicated by a specific unique identifier (here — the MitoDat ID number).

```
<UL>
<LI> <A HREF="/cgi-bin/dbEngine?mitoDat,getDataByID,MD100001">MD100001</A>
<LI> <A HREF="/cgi-bin/dbEngine?mitoDat,getDataByID,MD100005">MD100005</A>
<LI> <A HREF="/cgi-bin/dbEngine?mitoDat,getDataByID,MD100010">MD100010</A>
<LI> <A HREF="/cgi-bin/dbEngine?mitoDat,getDataByID,MD100015">MD100015</A>
    . . .
</UL>
```

2.5.4 Get a database entry by clicking on an object in an image — ismap

The fourth example illustrates how dbEngine is invoked from the HTML method using the HTML ISMAP protocol to return a specific entry using an image map. This appears to a user as a portion of Fig. 2.

```
Click on a gene label to return the corresponding database entry.
<A HREF="/cgi-bin/dbEngine/mitoDat,ismap,hum_01">
<A IMG SRC="hum_01.gif"> </A>
```

3 Results

We now illustrate dbEngine for the Web user with computer screens taken from an interactive session. The search results are shown in Figs. 4 and 5. Note the underlined hypertext references in the search results which point to other biomedical WWW databases. Figure 6 shows the list of database references which are synthesized as HTML forms from the raw reference database file.

4 Discussion

DbEngine is a WWW database search engine that accepts table data from computer spreadsheets or relational database software that makes it easy to publish simple databases on the Internet. Such data may be

easily maintained in the orginal spreadsheet database by office personnel. Data is exported from the spreadsheet program and copied to the dbEngine database file system as needed for updating the Web server. Because of dbEngine's Boolean search capability, data may be searched in more ways than the simple string searches that can be done on any document by a Web browser.

Since dbEngine may convert any data field to a hypertext link, it is easy to automatically generate hypertext links to other databases. Taking advantage of other WWW database servers cuts down duplication in a database by pointing to data maintained at other sites. By integrating interactive images into the dbEngine paradigm, a simple graphical user interface may be constructed to retrieve data directly by clicking on relevant objects in the images.

We created the MitoDat database because of the central role of the mitochondrion in the life of the cell. MitoDat is a repository of genome mapping and compartmental localization information, supplemented with reference data. Currently, MitoDat focuses on mammalian mitochondrial data, but it will be expanded to include mapping information from other species such as plant, yeast, and Drosophila. One use of this resource may be to draw new conclusions about synteny among disparate species in the nuclear genome. The database design is generic, to lay the foundation for a more general comparative
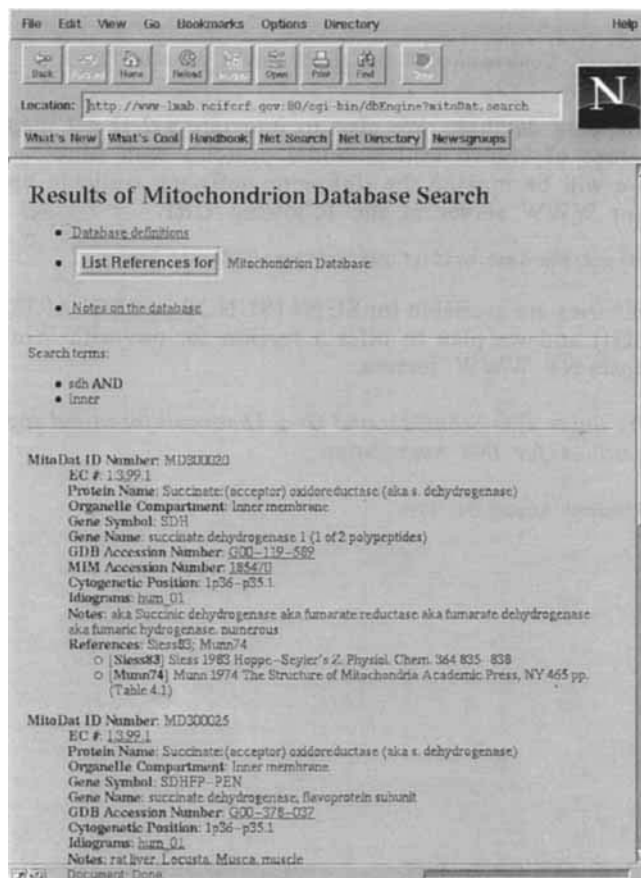

Figure 4. Search results of the search done in Fig. 3. Note the hypertext references in the results that point to other biomedical WWW databases.

Figure 5. Continuation of the search results done in Fig. 3.

Figure 6. List of database references that are synthesized as HTML forms from the raw reference database file.

mapping database. We plan to integrate a 2-D gel map image of known mitochondrial proteins with MitoDat. We will be making the dbEngine software available on our WWW server at the following URL

```
http://www-lmmb.ncifcrf.gov/Software/dbEngine.html
```

Binaries are available for SUN4 (SUNOS and SOLARIS, SGI) and we plan to offer a version for use with Windows-NT WWW servers.

5 References

[1] Berners-Lee, T. J., Cailliau, R., Luotonen, A., Henrick, F., Secret, A., World Wide Web. *Comm. Assoc. Comp. Mach.* 1994 *37(8)*76–82.

[2] Merril, C., Goldstein, M., Myrick, J., Creed, J., Lemkin, P. F., *Appl. Theor. Electrophoresis* 1995, *5*, 49–54.

[3] Lemkin, P. F., Orr, G. A., Goldstein, M., Creed, J., Myrick, J., Merril, C., *Appl. Theor. Electrophoresis* 1995, *5*, 55–72

[4] Wallace, D. C., *Annu. Rev. Biochem.* 1992, *61*, 1175–1212.

[5] Anderson, L., *Proc. Natl. Acad. Sci. USA* 1981, *78*, 2407–2411.

[6] Zuckerman, S. H., Linder, S., Eisenstadt, J. M., *Biochim. Biophys. Acta* 1984, *804*, 285–290.

[7] Zuckerman, S. H., Gillespie, F. P., Solus, J. F., Rybczynski, R., Eisenstadt, J. M., *Somat. Cell. Mol. Genet.* 1986, *12*, 449–458.