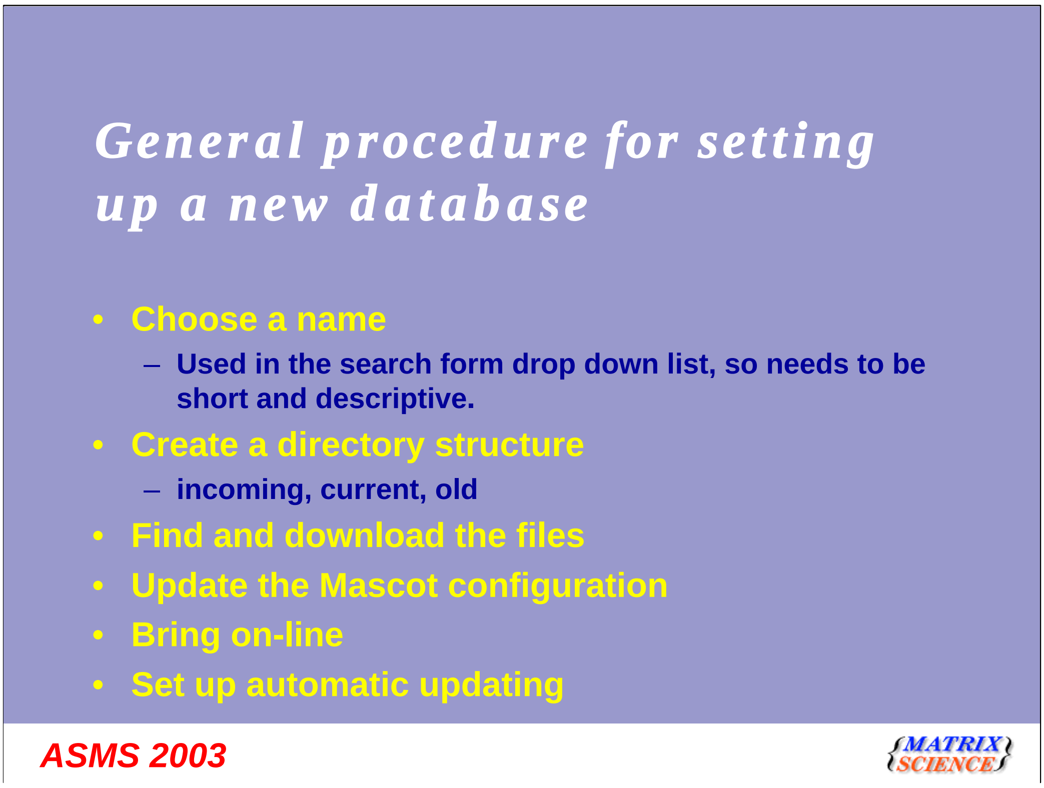
Sequence Database Download & Configuration

ASMS 2003

MATRIX SCIENCE

This talk will be mainly of interest to those people who administer an in-house Mascot server.



General procedure for setting up a new database

- Choose a name
  - Used in the search form drop down list, so needs to be short and descriptive.
- Create a directory structure
  - incoming, current, old
- Find and download the files
- Update the Mascot configuration
- Bring on-line
- Set up automatic updating

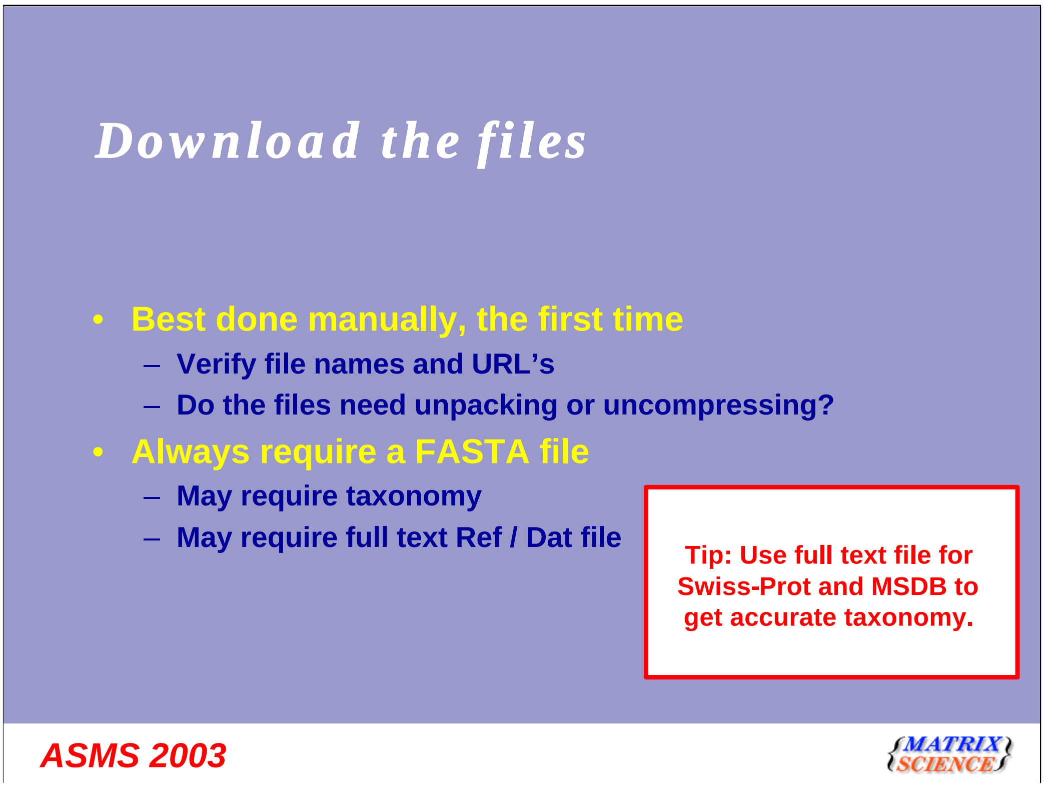
ASMS 2003

So, you've installed a copy of Mascot, which includes a copy of the MSDB protein database, and everything is working smoothly. Then you, or one of your colleagues, wants to search a different database. How do you go about adding a new database?

The general procedure is shown here:

<read from slide>

Let's focus on these last four steps in a little more detail



Download the files

- Best done manually, the first time
  - Verify file names and URL’s
  - Do the files need unpacking or uncompressing?
- Always require a FASTA file
  - May require taxonomy
  - May require full text Ref / Dat file

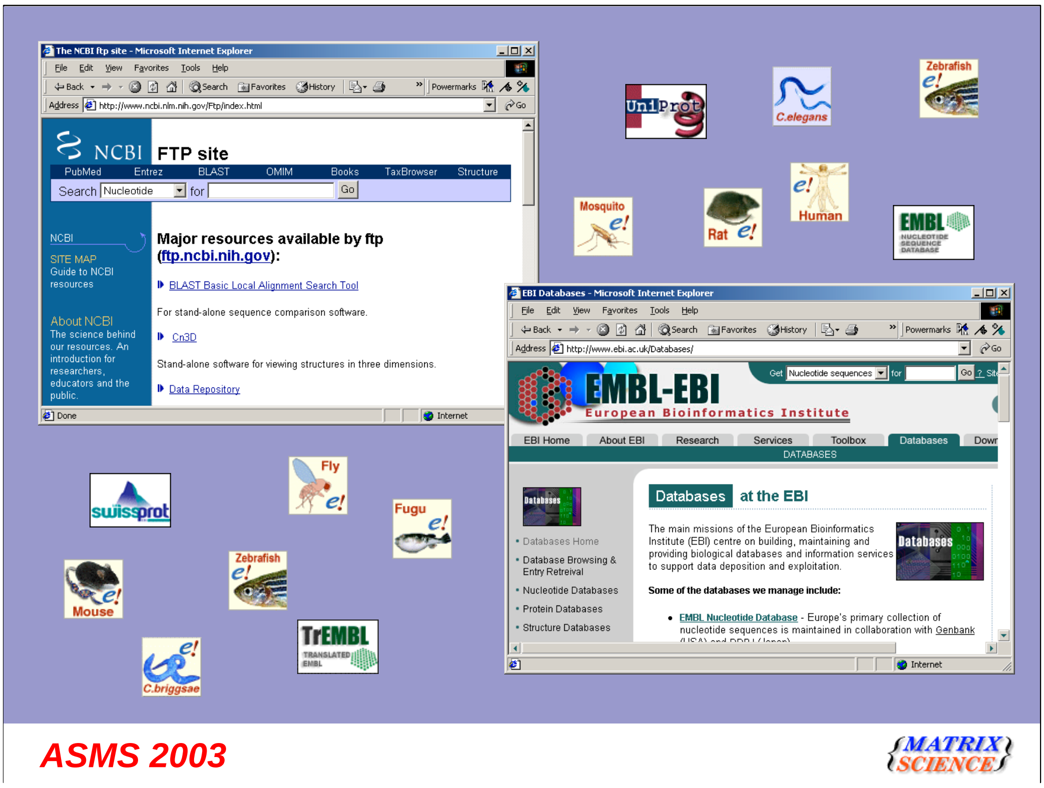
Tip: Use full text file for Swiss-Prot and MSDB to get accurate taxonomy.

ASMS 2003

First, catch your database, as Mrs Beeton would have said.

We'll talk later about automated downloading. But, when setting up a new database, you need to do everything manuallly at least once, to verify the filenames and URL's. If your running Mascot under Windows, you may find that you need to obtain utilities to decompress or untar the downloaded files.

At a minimum, we need a FASTA file. Some databases, such as Swiss-Prot and MSDB, also come with 'full text' files, containing comprehensive annotations. If a full text file is available and supported by Mascot, its a good idea to get it, if only because it will contain much more comprehensive taxonomy than can be included in a FASTA



FTP site

NCBI

SITE MAP
Guide to NCBI resources

About NCBI
The science behind our resources. An introduction for researchers, educators and the public.

Major resources available by ftp (ftp.ncbi.nih.gov):

- BLAST Basic Local Alignment Search Tool

For stand-alone sequence comparison software.

- Cn3D

Stand-alone software for viewing structures in three dimensions.

- Data Repository

Databases at the EBI

The main missions of the European Bioinformatics Institute (EBI) centre on building, maintaining and providing biological databases and information services to support data deposition and exploitation.

Some of the databases we manage include:

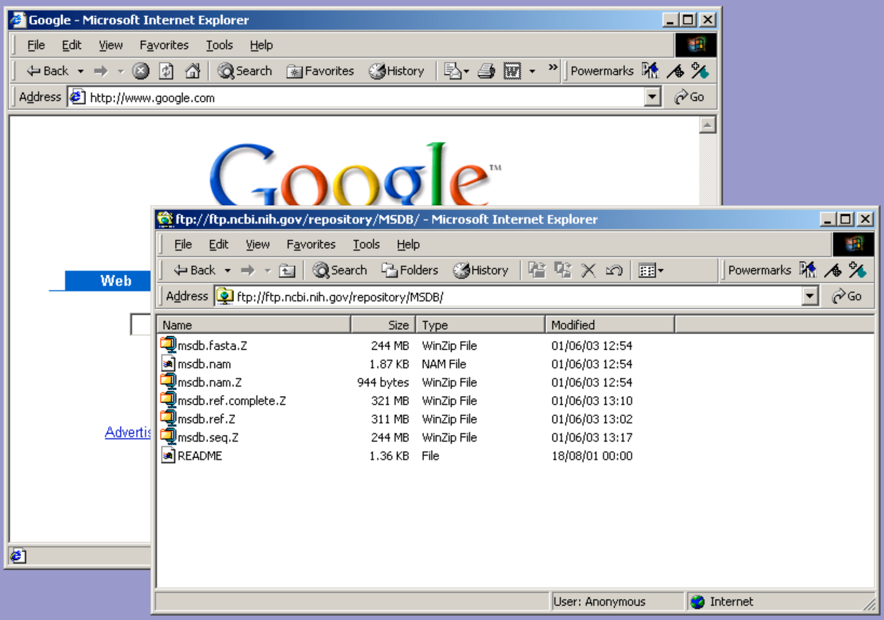
- EMBL Nucleotide Database - Europe's primary collection of nucleotide sequences is maintained in collaboration with Genbank

ASMS 2003

There are a few sites that provide comprehensive database repositiories. Two of the best known are NCBI and EBI.

Here, you can download nr, GenBank, Swiss-Prot, EMBL, Trembl, etc.

For specialised databases, such as individual genomes, you may have to track down the FTP site of the group that compiles the database



Name	Size	Type	Modified
msdb.fasta.Z	244 MB	WinZip File	01/06/03 12:54
msdb.nam	1.87 KB	NAM File	01/06/03 12:54
msdb.nam.Z	944 bytes	WinZip File	01/06/03 12:54
msdb.ref.complete.Z	321 MB	WinZip File	01/06/03 13:10
msdb.ref.Z	311 MB	WinZip File	01/06/03 13:02
msdb.seq.Z	244 MB	WinZip File	01/06/03 13:17
README	1.36 KB	File	18/08/01 00:00

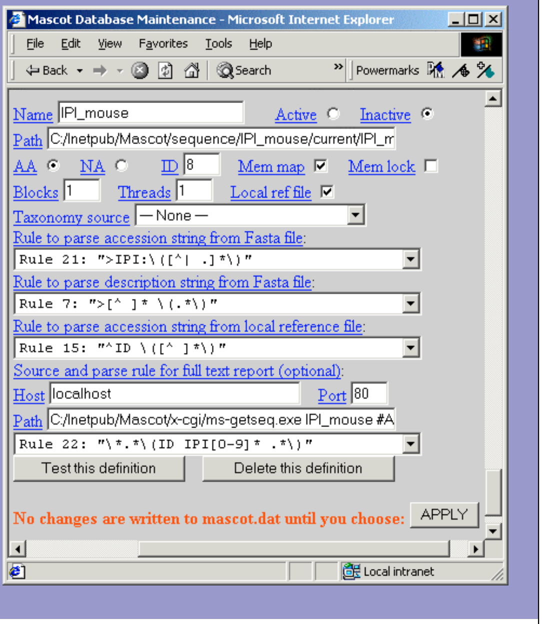
ASMS 2003

MATRIX SCIENCE

As with all web searches, a search engine such as Google is a good starting point. Once you find the database, you may have to study the README to figure out which files you need

Configuration

- **Database maintenance utility**
  - **Browser based**
- **Safer than editing mascot.dat**
- **Always creates a backup**
- **Simple test facility**



Name IPI_mouse Active ○ Inactive ●

Path C:/Inetpub/Mascot/sequence/IPI_mouse/current/IPI_m

AA ● NA ○ ID 8 Mem map ☑ Mem lock ☐

Blocks 1 Threads 1 Local ref file ☑

Taxonomy source — None —

Rule to parse accession string from Fasta file:

`Rule 21: ">IPI:\([^| .]*\)"`

Rule to parse description string from Fasta file:

`Rule 7: ">[^ ]* \(.*\)"`

Rule to parse accession string from local reference file:

`Rule 15: "^ID \([^ ]*\)"`

Source and parse rule for full text report (optional):

Host localhost Port 80

Path C:/Inetpub/Mascot/x-cgi/ms-getseq.exe IPI_mouse #A

`Rule 22: "\*.*\(ID IPI[0-9]* .*\)"`

No changes are written to mascot.dat until you choose:

So, having caught your database, you need to tell Mascot how to cook it.

We recommend that you use the browser based, database maintenance utility, rather than edit the configuration file directly. This reduces the chance of typos and ensures that there is always a backup in case of problems.

It also has a simple test facility, that will pick up most of the common configuration problems

FASTA Format

```
>Title text
SEQUENCESEQUENCESEQUENCESEQUENCESEQUENCESEQUENCE
SEQUENCESEQUENCESEQUENCESEQUENCESEQUENCESEQUENCE
SEQUENCESEQUENCESEQUENCE
>Next title
NEXTSEQUENCE …
```

```
>gi|6|bgi|Contig1.seq_7|2412 3299 [+3 L= 888] [Delayed

>20021010.2.1   1112073F09.y1 1112091F10.y1 1112073F0

>IPI:IPI00140097.1|REFSEQ_XP:XP_168061 Tax_Id=9606

>CCRB cytochrome c [validated] - rabbit

>gi|129249|sp|P02820|OSTC_BOVIN Osteocalcin precursor

>"ORF5 | start 2178-1309 | frame -1 | length=870 |
```

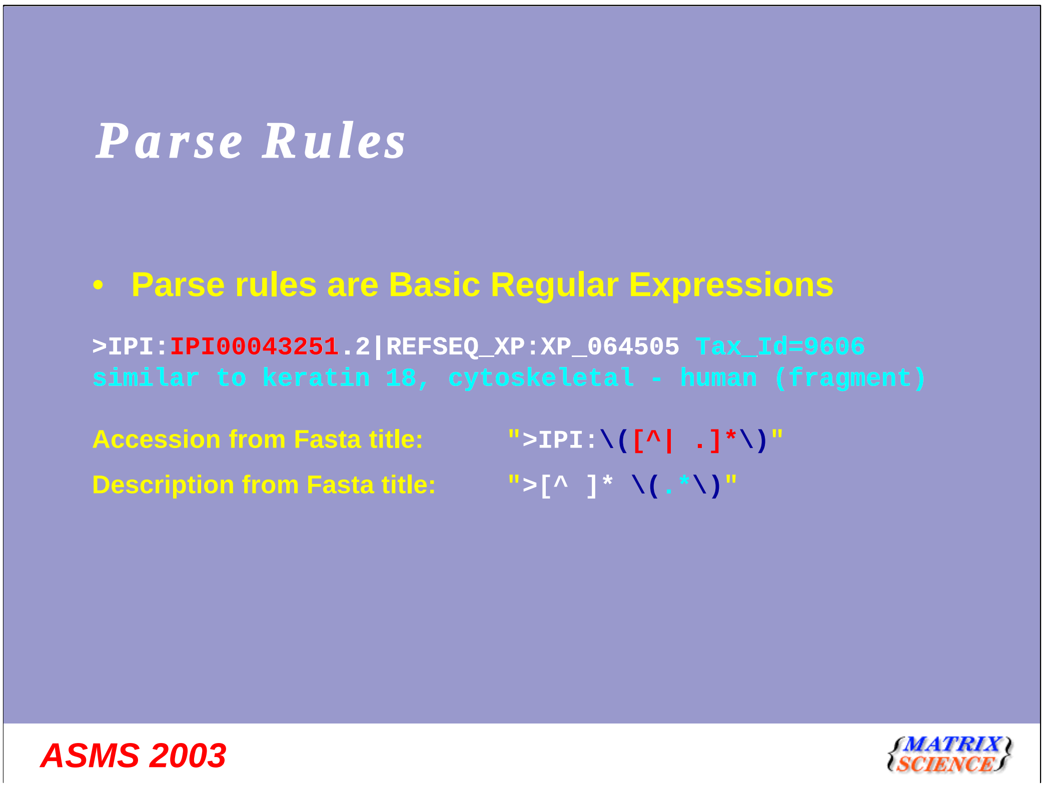
ASMS 2003

Perhaps this is a good moment to clarify exactly what we mean when we talk about a FASTA file.

FASTA is a very popular format. In a way, its like HTML ... it has become a universal standard because it is both very simple and very forgiving. On the down-side, it isn't much of a standard ... almost anything goes.

FASTA specifies that there will be a title line, starting with a 'greater than' character, followed by one or more lines containing the sequence in 1 letter code.

The problem is the lack of a well defined syntax for the title line. Here are a handful of examples of FASTA title lines. As you can see, there isn't much similarity. It is essential that we can find a short, unique identifier or accession string for each sequence. But, as you can see from these examples, the position of the identifier and the delimiters (e.g. spaces, pipe symbols, commas) varies considerably



Parse Rules

- Parse rules are Basic Regular Expressions

```
>IPI:IPI00043251.2|REFSEQ_XP:XP_064505 Tax_Id=9606
similar to keratin 18, cytoskeletal - human (fragment)
```

Accession from Fasta title:	`">IPI:\([^| .]*\)"`
Description from Fasta title:	`">[^ ]* \(.*\)"`

ASMS 2003

The way Mascot deals with this is to allow you to define the syntax for any particular database using a regular expression. Regular expression will be very familiar to anyone with a Unix background, but there may be a bit of a learning curve for someone with more of a Windows or Mac background.

Her, for example, we have a title line from the IPI human database. Lets say that we want to use IPI00043251 as the unique accession string and everything after the space should be treated as the description.

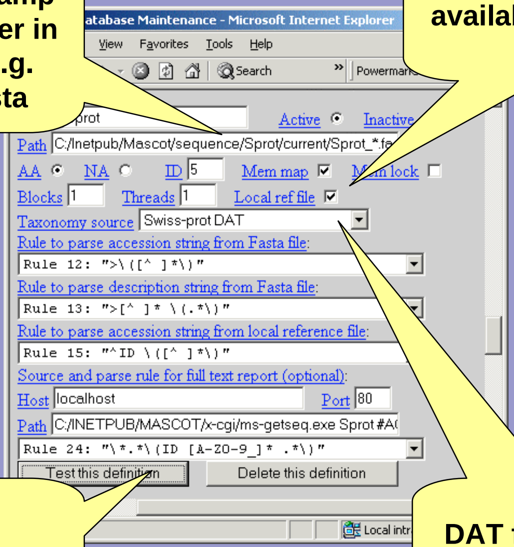
The regular expressions, or parse rules, used to extract this information look like this.

The string we want to extract is always within the backslashed parentheses. For the accession, we show the first few characters as literal text. We then say that we want to take all the following characters, stopping when we hit either a pipe symbol, a space, or a period. In fact, it is the period which applies in this example.

For the description, we discard everything up to and including the first space. Then, we take everything to the end of the title.

Include a timestamp or release number in the filename, e.g. Sprot_41.9.fasta

Full text DAT file is available for Swiss-Prot



prot
Active ● Inactive
Path C:/Inetpub/Mascot/sequence/Sprot/current/Sprot_*.fa
AA ● NA ○ ID 5 Mem map ☑ Mem lock ☐
Blocks 1 Threads 1 Local ref file ☑
Taxonomy source Swiss-prot DAT
Rule to parse accession string from Fasta file:

```
Rule 12: ">\([^ ]*\)"
```

Rule to parse description string from Fasta file:

```
Rule 13: ">[^ ]* \(.*\)"
```

Rule to parse accession string from local reference file:

```
Rule 15: "^ID \([^ ]*\)"
```

Source and parse rule for full text report (optional):
Host localhost Port 80
Path C:/INETPUB/MASCOT/x-cgi/ms-getseq.exe Sprot #A

```
Rule 24: "\*.*\(ID [A-Z0-9_]* .*\)"
```

TEST!

DAT file gives very accurate taxonomy

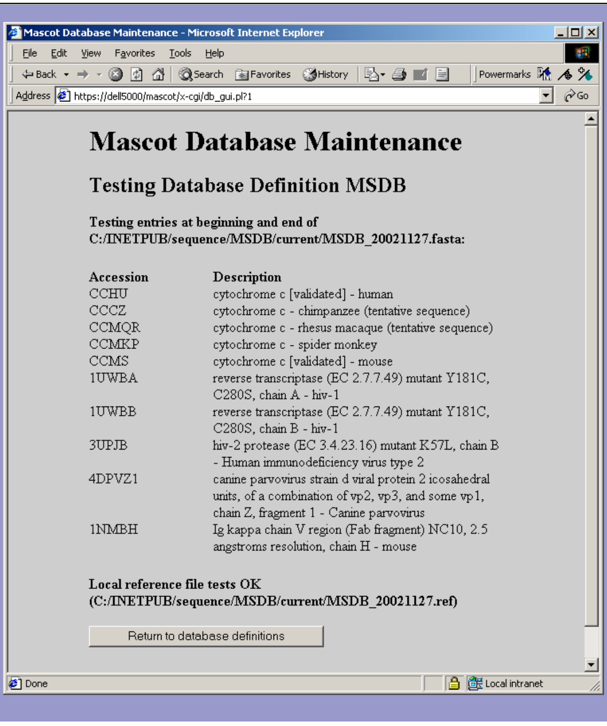
Deciding which parse rule to use, maybe even having to create a new parse rule, is definitely the most complicated part of configuring a new database. However, if we didn't use a tool like regular expressions, you would always be witing on us to add support into Mascot for new and unusual title formats.

The rest of the fields in database maintenance utility are pretty straight forward.

Its a good idea to include a version stamp or data stamp in the filename, so that you know what version of a database was used to obtain a particular result

In this example, Swiss-Prot, a full text DAT file is available, so we should make use of it to get accurate taxonomy

Once all the fields have been completed, hit the test button...



Mascot Database Maintenance

Testing Database Definition MSDB

Testing entries at beginning and end of C:/INETPUB/sequence/MSDB/current/MSDB_20021127.fasta:

Accession	Description
CCHU	cytochrome c [validated] - human
CCCZ	cytochrome c - chimpanzee (tentative sequence)
CCMQR	cytochrome c - rhesus macaque (tentative sequence)
CCMKP	cytochrome c - spider monkey
CCMS	cytochrome c [validated] - mouse
1UWBA	reverse transcriptase (EC 2.7.7.49) mutant Y181C, C280S, chain A - hiv-1
1UWBB	reverse transcriptase (EC 2.7.7.49) mutant Y181C, C280S, chain B - hiv-1
3UPJB	hiv-2 protease (EC 3.4.23.16) mutant K57L, chain B - Human immunodeficiency virus type 2
4DPVZ1	canine parvovirus strain d viral protein 2 icosahedral units, of a combination of vp2, vp3, and some vp1, chain Z, fragment 1 - Canine parvovirus
1NMBH	Ig kappa chain V region (Fab fragment) NC10, 2.5 angstroms resolution, chain H - mouse

Local reference file tests OK (C:/INETPUB/sequence/MSDB/current/MSDB_20021127.ref)

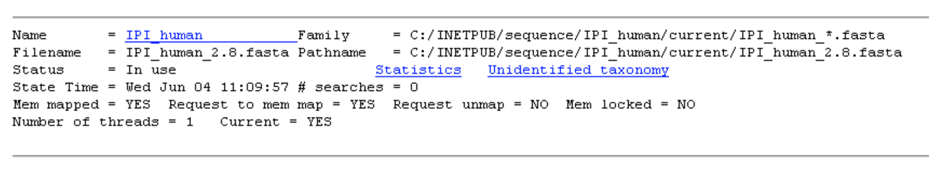
Return to database definitions

If you're unlucky, you'll see something like this. Looks like our first attempt at a parse rule wasn't quite right. OK, go back and try again

Success

Bring database on-line

- Monitor progress on the Database Status page
  - Errors may appear in Monitor log or Error log
  - Fix the problem and retry

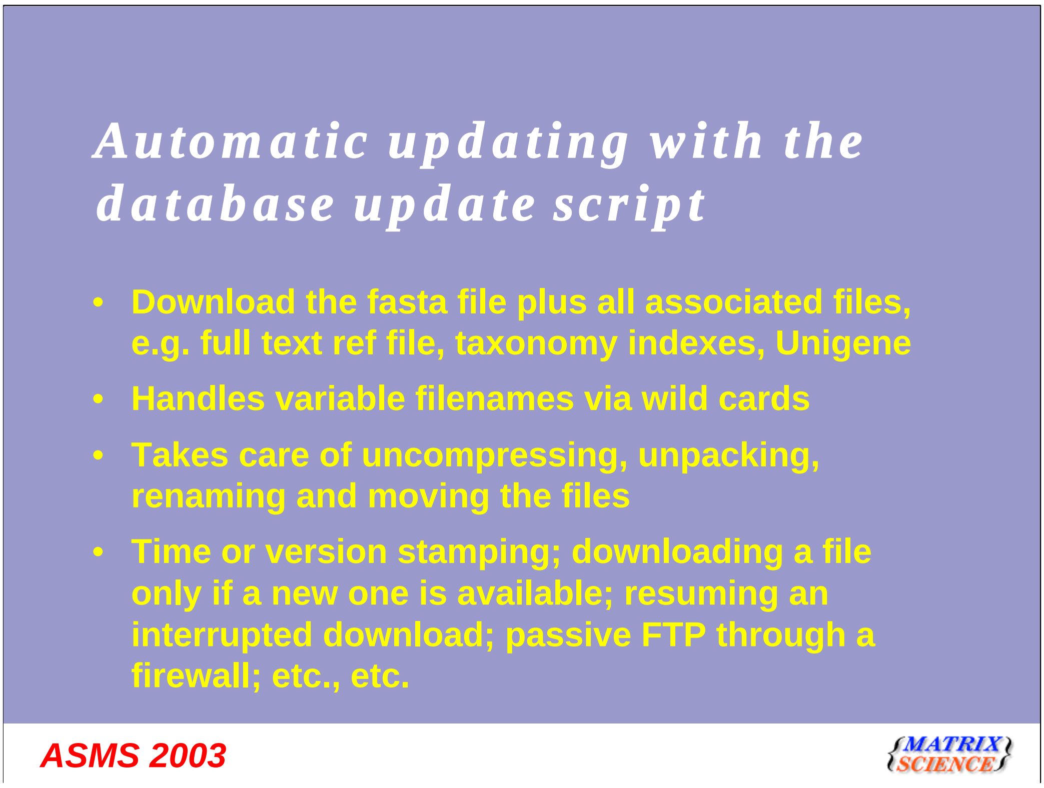


```
Name       = IPI_human              Family     = C:/INETPUB/sequence/IPI_human/current/IPI_human_*.fasta
Filename   = IPI_human_2.8.fasta Pathname   = C:/INETPUB/sequence/IPI_human/current/IPI_human_2.8.fasta
Status     = In use                        Statistics   Unidentified taxonomy
State Time = Wed Jun 04 11:09:57 # searches = 0
Mem mapped = YES  Request to mem map = YES  Request unmap = NO  Mem locked = NO
Number of threads = 1   Current = YES
```

Once the database tests successfully, we choose to apply the changes and bring the new database on-line

The test faciltiy can't pick up every problem. In particular, it only tests a few of the entries at the start of the database and a few at the end. So, its perfectly possible that something may go wrong in the middle.

If so, check the error log to see what the proble is, fix it, and re-try



Automatic updating with the database update script

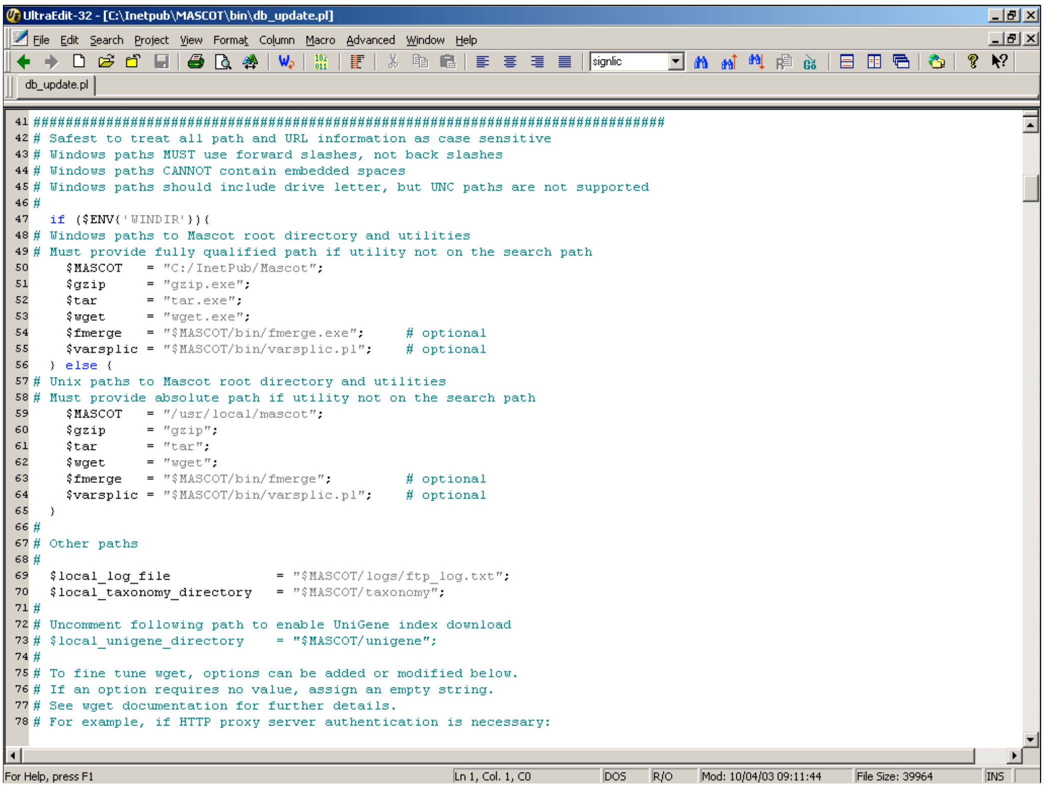
- Download the fasta file plus all associated files, e.g. full text ref file, taxonomy indexes, Unigene
- Handles variable filenames via wild cards
- Takes care of uncompressing, unpacking, renaming and moving the files
- Time or version stamping; downloading a file only if a new one is available; resuming an interrupted download; passive FTP through a firewall; etc., etc.

ASMS 2003

Now, our new database is ready for searching. What about updates? We don't want to have to remember to download updates manually every day or every week.

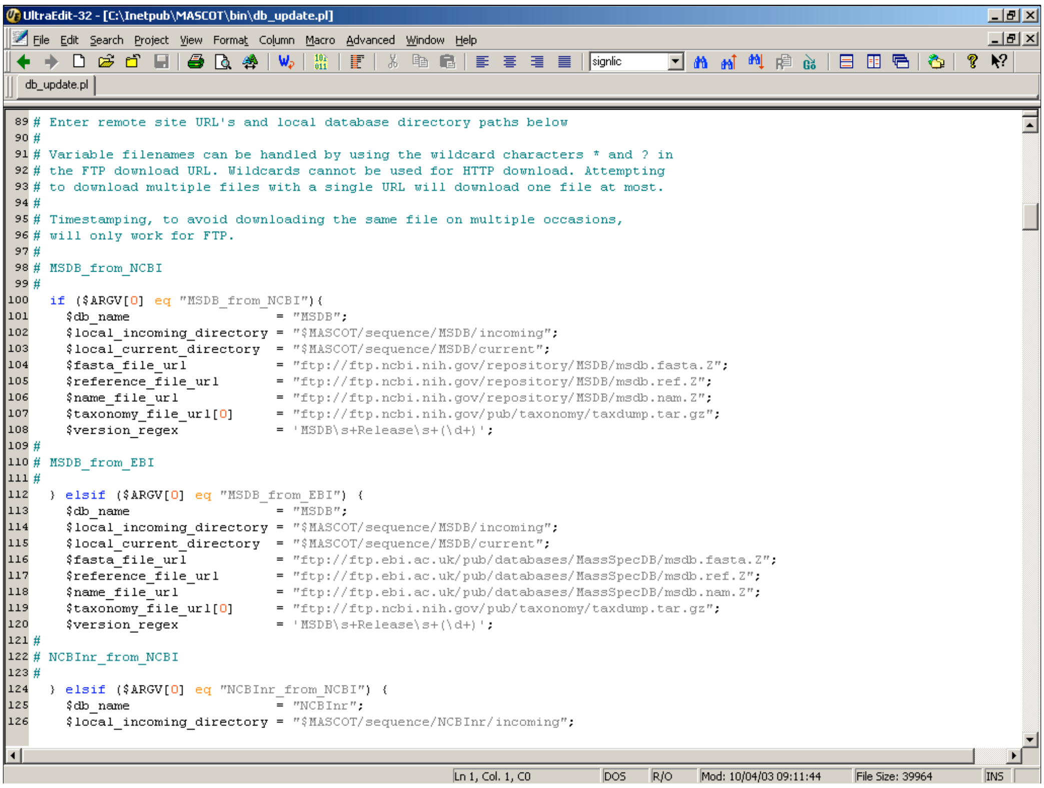
Mascot includes a script, the database update script, that will

<read from slide>



```
##############################################################################
# Safest to treat all path and URL information as case sensitive
# Windows paths MUST use forward slashes, not back slashes
# Windows paths CANNOT contain embedded spaces
# Windows paths should include drive letter, but UNC paths are not supported
#
  if ($ENV{'WINDIR'}){
# Windows paths to Mascot root directory and utilities
# Must provide fully qualified path if utility not on the search path
    $MASCOT   = "C:/InetPub/Mascot";
    $gzip     = "gzip.exe";
    $tar      = "tar.exe";
    $wget     = "wget.exe";
    $fmerge   = "$MASCOT/bin/fmerge.exe";     # optional
    $varsplic = "$MASCOT/bin/varsplic.pl";    # optional
  } else {
# Unix paths to Mascot root directory and utilities
# Must provide absolute path if utility not on the search path
    $MASCOT   = "/usr/local/mascot";
    $gzip     = "gzip";
    $tar      = "tar";
    $wget     = "wget";
    $fmerge   = "$MASCOT/bin/fmerge";         # optional
    $varsplic = "$MASCOT/bin/varsplic.pl";    # optional
  }
#
# Other paths
#
  $local_log_file             = "$MASCOT/logs/ftp_log.txt";
  $local_taxonomy_directory   = "$MASCOT/taxonomy";
#
# Uncomment following path to enable UniGene index download
# $local_unigene_directory    = "$MASCOT/unigene";
#
# To fine tune wget, options can be added or modified below.
# If an option requires no value, assign an empty string.
# See wget documentation for further details.
# For example, if HTTP proxy server authentication is necessary:
```

The database update script is a perl script. If you've installed Mascot into the default path, you should be able to use this script as is to download updates for many widely used databases.

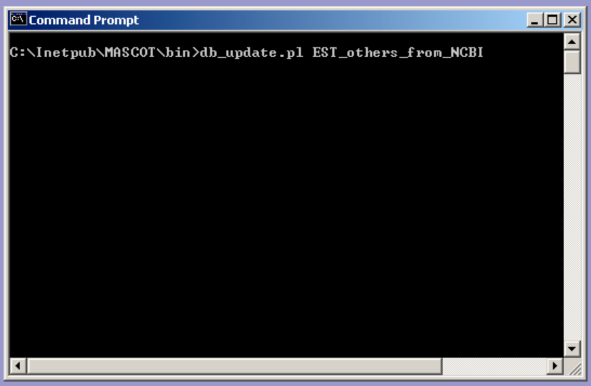


```
# Enter remote site URL's and local database directory paths below
#
# Variable filenames can be handled by using the wildcard characters * and ? in
# the FTP download URL. Wildcards cannot be used for HTTP download. Attempting
# to download multiple files with a single URL will download one file at most.
#
# Timestamping, to avoid downloading the same file on multiple occasions,
# will only work for FTP.
#
# MSDB_from_NCBI
#
  if ($ARGV[0] eq "MSDB_from_NCBI"){
    $db_name                  = "MSDB";
    $local_incoming_directory = "$MASCOT/sequence/MSDB/incoming";
    $local_current_directory  = "$MASCOT/sequence/MSDB/current";
    $fasta_file_url           = "ftp://ftp.ncbi.nih.gov/repository/MSDB/msdb.fasta.Z";
    $reference_file_url       = "ftp://ftp.ncbi.nih.gov/repository/MSDB/msdb.ref.Z";
    $name_file_url            = "ftp://ftp.ncbi.nih.gov/repository/MSDB/msdb.nam.Z";
    $taxonomy_file_url[0]     = "ftp://ftp.ncbi.nih.gov/pub/taxonomy/taxdump.tar.gz";
    $version_regex            = 'MSDB\s+Release\s+(\d+)';
#
# MSDB_from_EBI
#
  } elsif ($ARGV[0] eq "MSDB_from_EBI") {
    $db_name                  = "MSDB";
    $local_incoming_directory = "$MASCOT/sequence/MSDB/incoming";
    $local_current_directory  = "$MASCOT/sequence/MSDB/current";
    $fasta_file_url           = "ftp://ftp.ebi.ac.uk/pub/databases/MassSpecDB/msdb.fasta.Z";
    $reference_file_url       = "ftp://ftp.ebi.ac.uk/pub/databases/MassSpecDB/msdb.ref.Z";
    $name_file_url            = "ftp://ftp.ebi.ac.uk/pub/databases/MassSpecDB/msdb.nam.Z";
    $taxonomy_file_url[0]     = "ftp://ftp.ncbi.nih.gov/pub/taxonomy/taxdump.tar.gz";
    $version_regex            = 'MSDB\s+Release\s+(\d+)';
#
# NCBInr_from_NCBI
#
  } elsif ($ARGV[0] eq "NCBInr_from_NCBI") {
    $db_name                  = "NCBInr";
    $local_incoming_directory = "$MASCOT/sequence/NCBInr/incoming";
```

Here you see the section of the script where we define which files are needed for a particular database. The script takes care of downloading the FASTA and full text files, plus any taxonomy files that are required. The final line of the configuration describes how to find the database version number in the release notes. Another regular expression!

Predefined configurations

- MSDB
- NCBInr
- EST_human
- EST_mouse
- EST_others
- Sprot
- Trembl
- IPI_human
- IPI_mouse

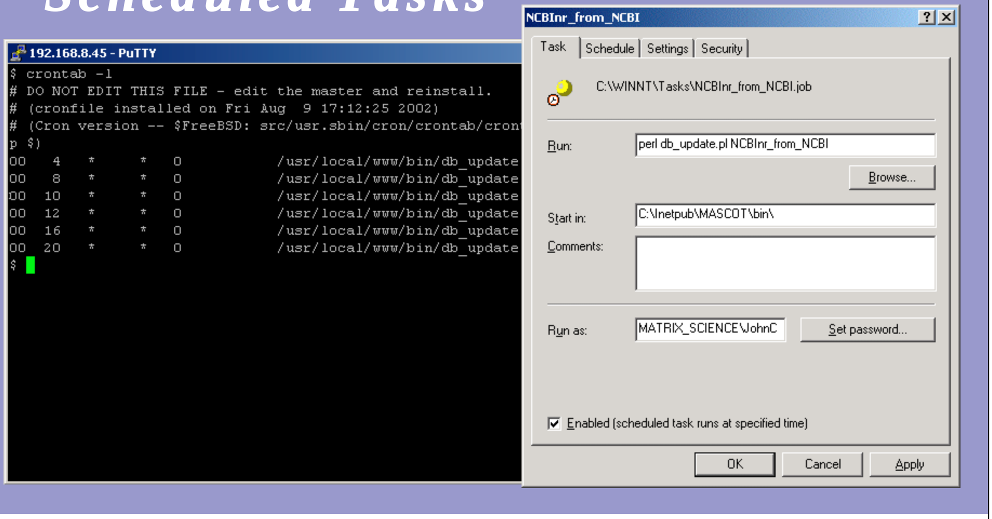


```
C:\Inetpub\MASCOT\bin>db_update.pl EST_others_from_NCBI
```

All these databases are predefined in the script, so to update all the files, you only need to type something like this at a command or shell prompt.

Adding a new database to the script is quite straight forward, just use one of these as a template

Automate with Cron or Scheduled Tasks



```
$ crontab -l
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (cronfile installed on Fri Aug  9 17:12:25 2002)
# (Cron version -- $FreeBSD: src/usr.sbin/cron/crontab/cront
p $)
00   4   *     *   0            /usr/local/www/bin/db_update
00   8   *     *   0            /usr/local/www/bin/db_update
00  10   *     *   0            /usr/local/www/bin/db_update
00  12   *     *   0            /usr/local/www/bin/db_update
00  16   *     *   0            /usr/local/www/bin/db_update
00  20   *     *   0            /usr/local/www/bin/db_update
$ 
```

C:\WINNT\Tasks\NCBInr_from_NCBI.job

Run: perl db_update.pl NCBInr_from_NCBI

Start in: C:\Inetpub\MASCOT\bin\

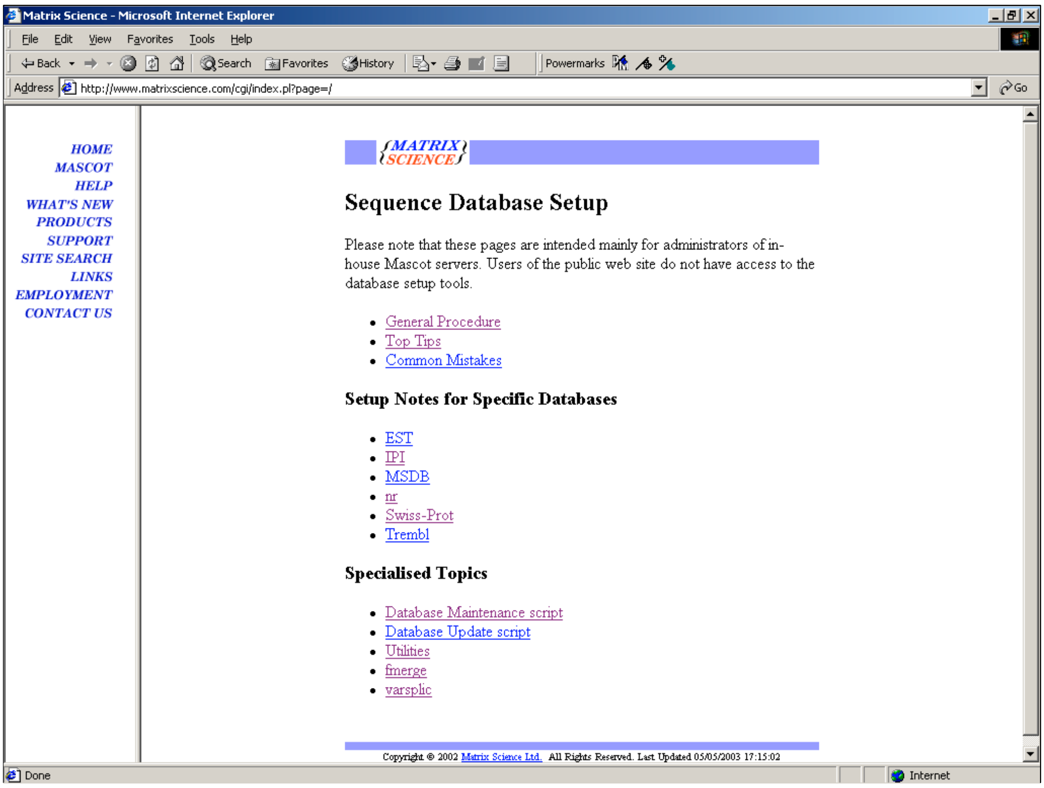
Comments:

Run as: MATRIX_SCIENCE\JohnC

☑ Enabled (scheduled task runs at specified time)

Now, we are in a poition to automate the whole process. Under Unix, we use Cron to execute the script on a pre-defined schedule

Under Windows, we use Scheduled Tasks



Sequence Database Setup

Please note that these pages are intended mainly for administrators of in-house Mascot servers. Users of the public web site do not have access to the database setup tools.

- General Procedure
- Top Tips
- Common Mistakes

Setup Notes for Specific Databases

- EST
- IPI
- MSDB
- nr
- Swiss-Prot
- Trembl

Specialised Topics

- Database Maintenance script
- Database Update script
- Utilities
- fmerge
- varsplic

Finally, we have made an effort to provide detailed walk-throughs for configuring all the common databases. Please refer to the on-line help for greater detail on all the topics I've mentioned here