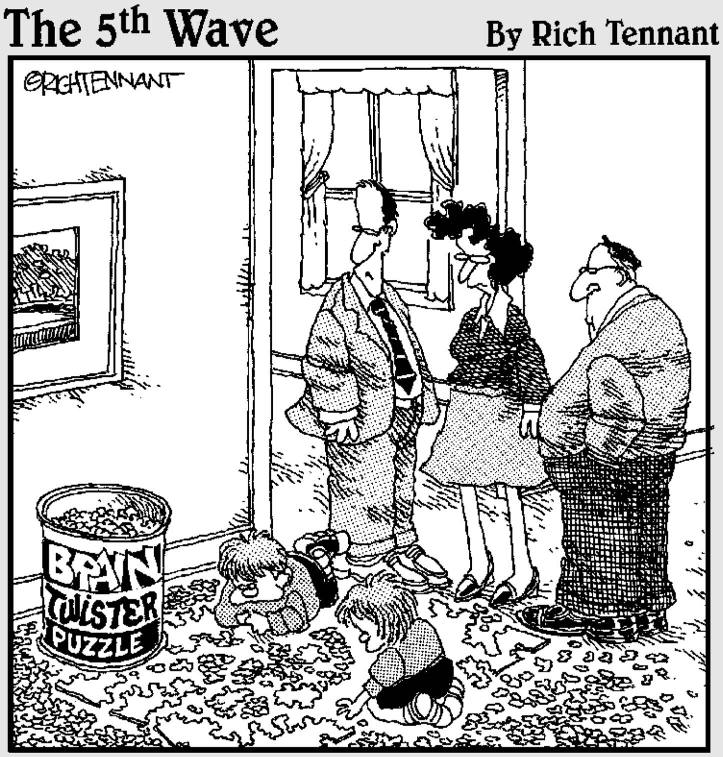
in Sequence Analysis



"That reminds me. I have to do some multiple sequence alignments later on."

In this part . . .

The most powerful tools used in bioinformatics rely on sequence comparison methods. In this part, we show you how you can search a database by sequence comparison using BLAST — or how you can use pairwise comparison techniques (such as Dotlet) to compare two sequences. We also show you multiple alignment programs — such as ClustalW or Tcoffee — that compare many sequences at the same time. If all these things are new to you, beware: They'll forever change the way you do biology!

Similarity Searches on Sequence Databases

In This Chapter

- Discovering why similarity is important and what it can do for you
- Finding out everything you need to know about homology
- Running BLAST over the Internet to find a sequence similar to yours
- Determining whether your result means something
- Choosing the right flavor of BLAST
- Looking for remote homologues with PSI-BLAST
- Discovering new protein domains with PSI-BLAST

> *"When looking for a needle in a haystack, the optimistic wears gloves."*
>
> — The Little Book of Things to Keep in Mind

If you already have a protein or DNA sequence and you want to find other sequences that look like it, then you've come to the right place. In this chapter, we show you how to use BLAST to compare your sequence with every sequence contained in a sequence database — and keep the best results — in two clicks of a mouse. We also show you situations where you can use PSI-BLAST, a more powerful version of BLAST, to ask biological questions.

On the other hand, if you only know the name of your sequence or if you can only describe this sequence with words (*mouse serine and protease,* for instance) — rather than with its actual amino-acid sequence — then the first step is to go and look for this sequence in a database. We explain how to use names or descriptions to look for a sequence in Chapter 2.

similar sequences often derive from the same ancestral sequence. This means that if your sequences are similar, they probably have the same ancestor, share the same structure, and have a similar biological function. This principle even works when the sequences come from very different organisms. For you, this means you can extrapolate something you know about a particular DNA or protein sequence to all similar DNA and protein sequences.

For example, imagine that your favorite sequence looks very much like another one that somebody has studied in another lab. Because these two sequences are similar, you can say, "If something is true for that sequence, it is probably true for mine as well!" Just imagine how much time you can save: Studying a gene in the lab takes years; searching a database for similarity takes seconds. And you're not even cheating!

When two proteins or gene sequences are very similar, biologists call them *homologues,* which is a fancy word for two proteins or gene sequences that have the same ancestor, similar functions, and similar structures. The snag comes in deciding how similar is "very" similar. If your sequences are more than 100 amino acids long (or 100 nucleotides long), the rule says you can label proteins as "homologous" if 25 percent of the amino acids are identical, for DNA you will require at least 70 percent identity to draw the same conclusion. If your values are below those stated values, then your guess is as good as any. You've entered that range of protein identity below 25 percent — known (fans of Rod Serling take note) as the *twilight zone* — where

- **Nothing is sure about the meaning of observed similarities.**

  For instance, some proteins whose amino acids are less than 15 percent identical have exactly the same 3-D structure — while some proteins with residues that are 20 percent identical have different structures.

- **Homology or non-homology is never granted.**

The 25-percent figure that defines the twilight zone is mostly a common-sense indicator. In reality, things are slightly more complicated. In most cases, to make sure that two sequences are true homologues, you also need to use some other information reported by the search. These bits of data include

- The *Expectation value* (E-value), which tells you how likely it is that the similarity between your sequence and a database sequence is due to chance
- The length of the segments similar between the two sequences

are either homologous or non-homologous — while they will always have a measurable level of identity. As a consequence, you cannot say that two sequences are 40 percent homologous, just like you cannot say someone is 40 percent pregnant.

In the rest of this chapter, we show you how to interpret this information when you conduct a database search.

The Most Popular Data-Mining Tool Ever: BLAST

Thirty years ago, biologists would search a database by printing its whole content on paper, taping the printout to the office wall, writing down their own query sequence on a piece of paper, and spending a few hours manually scanning the wall and drinking coffee. With the millions of sequences now available, those days are gone. Fortunately, now you can use computers to run the most successful bioinformatics tool ever: BLAST, the *B*asic *L*ocal *A*lignment and *S*earch *T*ool, which is expressly designed to do the paper-on-the-wall scanning for you.

In this section, we show you two simple ways to do a BLAST search (by using a protein or a DNA sequence), and, most importantly, we show you how to interpret your results. At the end of the section, we also give you plenty of ideas on how to use BLAST for doing most of the things you could need (except making coffee).

BLASTing protein sequences

BLASTing protein sequences is what you want to do if you already have a protein sequence and you want to find other, similar protein sequences in a sequence database. You should be happy to know that BLAST is at its best with proteins. Two flavors of BLAST that exist and deal with proteins are

- **blastp:** Compares a protein sequence with a protein database.
- **tblastn:** Compares a protein sequence with a nucleotide database.

Table 7-1 Choosing the Right BLAST Flavor for Proteins

What You Want	*The Right BLAST Flavor*
I want to find out something about the function of my protein.	Use blastp to compare your protein with other proteins contained in the databases.
I want to discover new genes encoding simple proteins.	Use tblastn to compare your protein with DNA sequences translated into their six possible reading frames (three on each strand).

In this section, we show you how to use two of the most popular blastp online services:

- The blastp server from its home, the National Center for Biotechnology Information (NCBI) in the USA
- The blastp server from the Swiss EMBnet server

These two servers have slightly different interfaces. If you know how to use them both, you're sure to feel at ease using any of the BLAST servers that we list at the end of this chapter.

TIP

Two good reasons for knowing how to use many BLAST servers are

- **The databases:** The BLAST servers don't give you access to the same databases. If you don't find the database you need on one server, you can always look for it on another server.
- **The speed:** The most popular servers are often overcrowded. When this happens, searching somewhere else is always an option.

WARNING!

It is a rarity for two different BLAST servers to return the same answer when you do the same query. The main reason is that the database versions often differ a little, although differences in the BLAST program version can have an effect as well. Results rarely change dramatically, but they do change a little. It's just one of those discrepancies that you must learn to live with.

Running the NCBI blastp

In this example, you are asking the following question: "Are there any proteins similar to the hamster nucleolin in the protein database Swiss-Prot?" Good question. Here's how you find the answer:

A search page similar to the one in Figure 7-2 appears.

3. **Paste your sequence in the search window.**

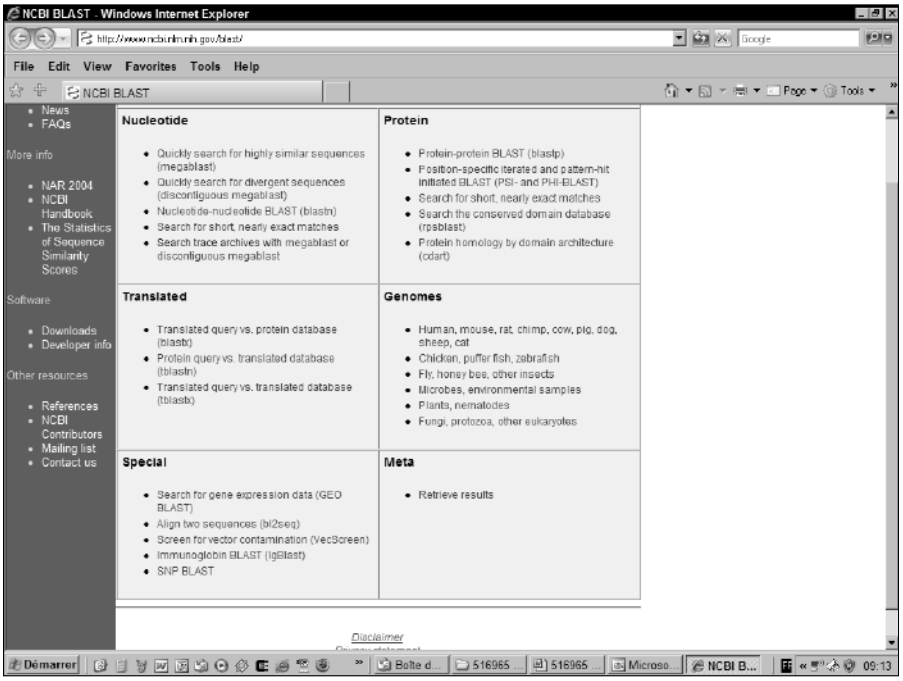
   You have two ways to pass on a sequence to the blastp server:

   - If your sequence is already in a database, you can give its ID number to blastp. For our example, we use the hamster *nucleolin,* a protein from Swiss-Prot whose accession number is P09405. To use the nucleolin, enter **P09405** in the sequence box, as shown in Figure 7-2.
   - If your sequence is not in a database, you must provide it in the FASTA format. Figure 7-3 shows you what the hamster nucleoline sequence looks like in FASTA format.

   The sequence you give to blastp is the *query sequence.*

   Sequences similar to the query that blastp returns are the *hits* or *matches.*

   The database you search is the *target database.*



NCBI BLAST - Windows Internet Explorer

http://www.ncbi.nlm.nih.gov/blast/

File Edit View Favorites Tools Help

NCBI BLAST

- News
- FAQs

More info

- NAR 2004
- NCBI Handbook
- The Statistics of Sequence Similarity Scores

Software

- Downloads
- Developer info

Other resources

- References
- NCBI Contributors
- Mailing list
- Contact us

Nucleotide

- Quickly search for highly similar sequences (megablast)
- Quickly search for divergent sequences (discontiguous megablast)
- Nucleotide-nucleotide BLAST (blastn)
- Search for short, nearly exact matches
- Search trace archives with megablast or discontiguous megablast

Protein

- Protein-protein BLAST (blastp)
- Position-specific iterated and pattern-hit initiated BLAST (PSI- and PHI-BLAST)
- Search for short, nearly exact matches
- Search the conserved domain database (rpsblast)
- Protein homology by domain architecture (cdart)

Translated

- Translated query vs. protein database (blastx)
- Protein query vs. translated database (tblastn)
- Translated query vs. translated database (tblastx)

Genomes

- Human, mouse, rat, chimp, cow, pig, dog, sheep, cat
- Chicken, puffer fish, zebrafish
- Fly, honey bee, other insects
- Microbes, environmental samples
- Plants, nematodes
- Fungi, protozoa, other eukaryotes

Special

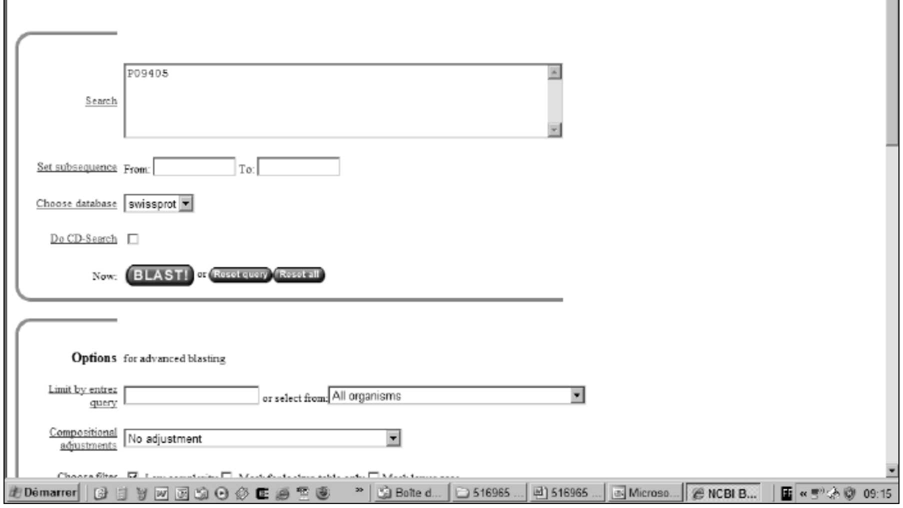
- Search for gene expression data (GEO BLAST)
- Align two sequences (bl2seq)
- Screen for vector contamination (VecScreen)
- Immunoglobin BLAST (IgBlast)
- SNP BLAST

Meta

- Retrieve results

Disclaimer

Figure 7-1: The BLAST home page at NCBI.



Search: P09405

Set subsequence From: To:

Choose database: swissprot

Do CD-Search □

Options for advanced blasting

Limit by entrez query: or select from: All organisms

Compositional adjustments: No adjustment

Figure 7-2: The blastp server at NCBI.

4. **Choose Swiss-Prot from the Choose Database pull-down menu.**

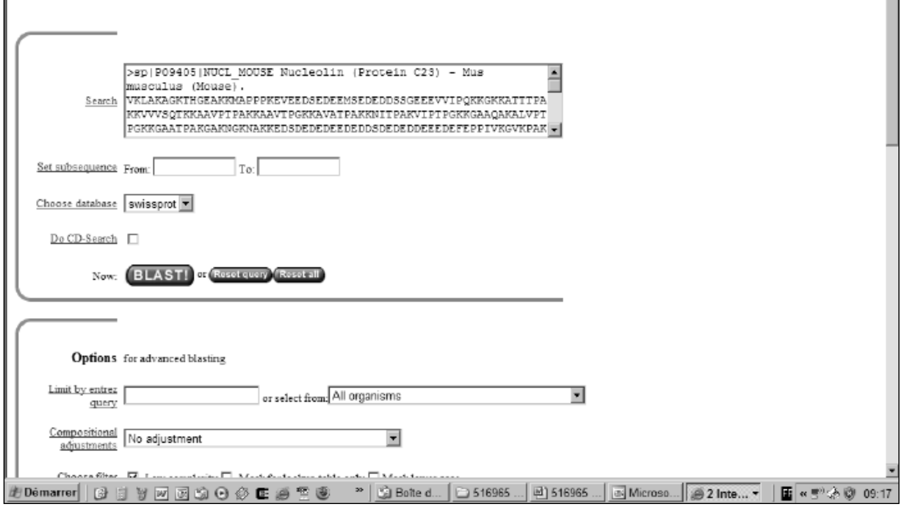
5. **Deselect the Do CD-Search box.**

   CD search is a feature for searching *C*onserved *D*omains. We recommend you deselect this box if this is one of your first BLAST searches because the output may be confusing. (We explain CD search in detail in Chapter 6.)

6. **Click the BLAST! button (and wait).**

   An intermediate page similar to Figure 7-4 appears.

   Sometimes the NCBI BLAST server gets very busy. For instance, it is notorious for scientists to dump a large number of jobs just before they go home and let them run at night. This explains why BLAST can be uncomfortable to use in the evening. Fortunately, NCBI is not the only BLAST server around; you can always try your luck on another server. (See Table 7-6, at the end of this chapter, for a listing of BLAST servers around the world.)



Search
```
>sp|P09405|NUCL_MOUSE Nucleolin (Protein C23) - Mus
musculus (Mouse).
VKLAKAGKTHGEAKKMAPPPKEVEEDSEDEEMSEDEDDSSGEEEVVIPQKKGKKATTTPA
KKVVVSQTKKAAVPTPAKKAAVTPGKKAVATPAKKNITPAKVIPTPGKKGAAQAKALVPT
PGKKGAATPAKGAKNGKNAKKEDSDEDEDEEDEDDSDEDEDDEEEDEFEPPIVKGVKPAK
```

Set subsequence From: To:

Choose database swissprot

Do CD-Search □

Now: BLAST! or Reset query Reset all

Options for advanced blasting

Limit by entrez query or select from: All organisms

Compositional adjustments No adjustment

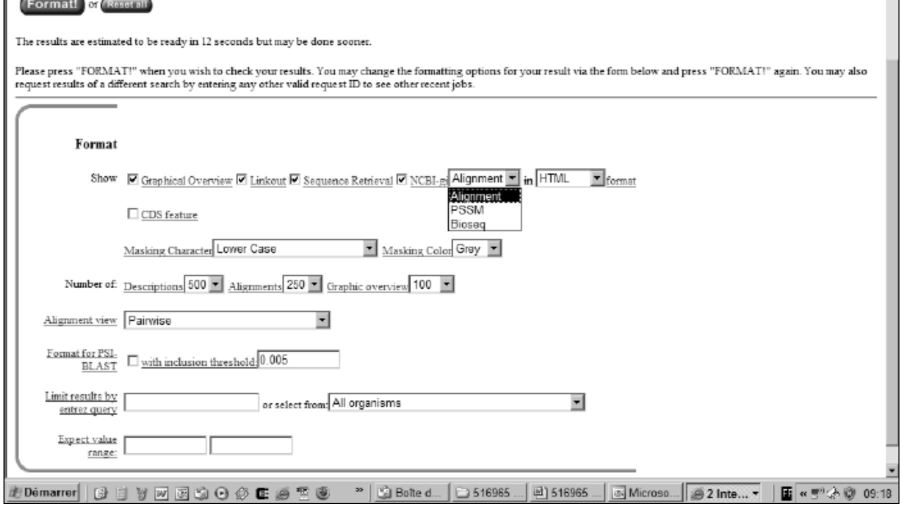
Figure 7-3: Providing the NCBI server with a sequence in FASTA format.

If the server nearest you is too slow, you can take advantage of the world's various time zones. The following table indicates the part of the world that's sleeping while you do your work in the morning or in the afternoon.

Your Location	*Morning*	*Afternoon*
USA	Japan	Europe
Europe	USA	Japan
Japan	Europe	USA

7. **Click the Format! button on the intermediate page and wait for the results.**

   When you click the Format! button, a new browser window opens. As soon as the search is complete, BLAST displays your results in this new window. Unfortunately, the search may take more time than the form indicates.



The results are estimated to be ready in 12 seconds but may be done sooner.

Please press "FORMAT!" when you wish to check your results. You may change the formatting options for your result via the form below and press "FORMAT!" again. You may also request results of a different search by entering any other valid request ID to see other recent jobs.

Format

Show ☑ Graphical Overview ☑ Linkout ☑ Sequence Retrieval ☑ NCBI-g Alignment in HTML format

Alignment
PSSM
Bioseq

☐ CDS feature

Masking Character Lower Case Masking Color Grey

Number of: Descriptions 500 Alignments 250 Graphic overview 100

Alignment view Pairwise

Format for PSI-BLAST ☐ with inclusion threshold 0.005

Limit results by entrez query or select from: All organisms

Expect value range:

Figure 7-4: The BLAST intermediate result page.

A typical search against NR or Swiss-Prot takes a few minutes. If BLAST tells you that the completion of your search may take more than 20 minutes, you're probably better off trying your luck some other time — or on another BLAST server.

DO NOT press any button while you wait!

If you get no reply, DO NOT resubmit the same query several times in a row — it will only make things worse for everybody (including you)!

8. **Save your results.**

Do not try to print the results; the alignment section may be HUGE. If you want to keep a trace of this search, save it in a file by using the File➪Save As option of your browser.

Unfortunately, saving a complete NCBI BLAST page isn't easy. Saved pages usually can't redisplay their active graphics when you reload them in your browser. Here are two (still rather unsatisfactory) solutions:

- **Save the graphic display:** Pass the mouse over the graphic display, right-click, and then choose Save Picture As from the context menu that appears, as shown in Figure 7-5. The saved image is in GIF format. Although you can then include the display in electronic documents, be aware that all active hyperlinks are now lost.

The BLAST server running on the EMBnet-ExPASy server is similar to the one running at NCBI, but it uses a different interface — very similar to the one used by many BLAST servers around the world. If you know how to use the EMBnet blastp, you know how to use almost any blastp server in the world.

The main difference between the NCBI blastp interface and the EMBnet interface (shown in Figure 7-6) is that the EMBnet blastp interface gives you many more choices. The downside is that you are the one who has to make sure that the various selected parameters are compatible. In the following steps, we show you how to avoid any confusion.

1. **Point your browser to** `www.ch.embnet.org/software/bBLAST.html`.

   The EMBnet Basic BLAST page appears.

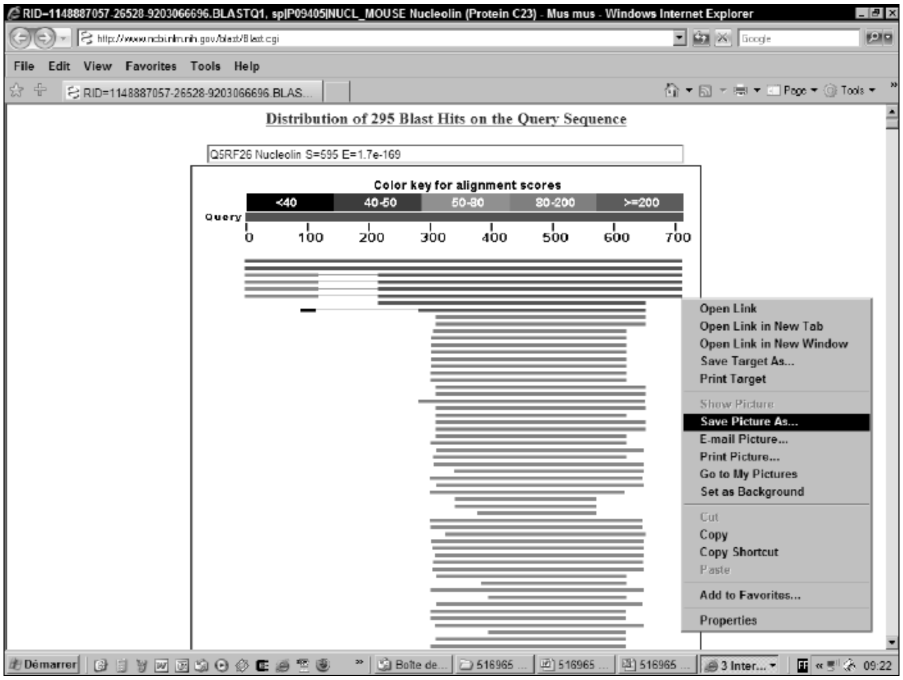
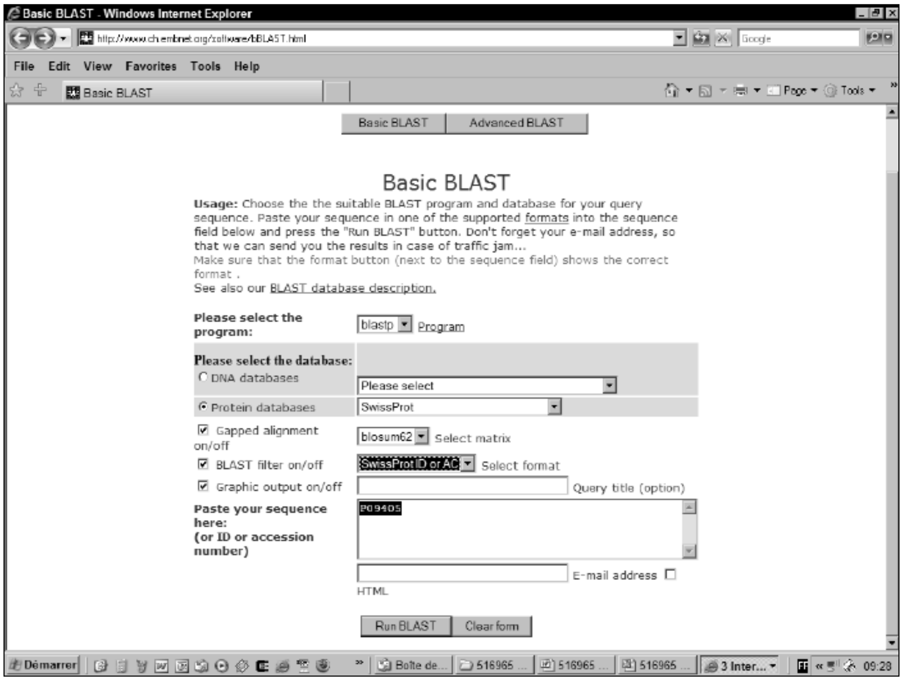


Figure 7-5: Saving the NCBI graphic display.

The database you select must be compatible with the BLAST program that you choose in Step 2; the EMBnet server does not check this for you. The kind of database you can search with a protein sequence as a query depends on which flavor of BLAST you selected in Step 2:

BLAST flavor	***Database type***
blastp,blastx	Protein
blastn,tblastn,tblastx	DNA

Selecting a genome database can make it easier to analyze your results. Genome databases are those with a species name and the word *genome* in their names, such as *C.Elegans Genome.* This can be especially useful if the subject of your query belongs to a very large family. If you know which species you're interested in, choose it there. If you can't find it, use the advanced version of this server at `www.ch.embnet.org/software/aBLAST.html`.



Basic BLAST

Usage: Choose the the suitable BLAST program and database for your query sequence. Paste your sequence in one of the supported formats into the sequence field below and press the "Run BLAST" button. Don't forget your e-mail address, so that we can send you the results in case of traffic jam...
Make sure that the format button (next to the sequence field) shows the correct format .
See also our BLAST database description.

Please select the program: blastp Program

Please select the database:
○ DNA databases: Please select
● Protein databases: SwissProt

☑ Gapped alignment on/off — blosum62 Select matrix
☑ BLAST filter on/off — SwissProt ID or AC Select format
☑ Graphic output on/off — Query title (option)

Paste your sequence here: (or ID or accession number) P09405

E-mail address □ HTML

Figure 7-6: The EMBnet BLAST interface.

format without the first line (the one that starts with >). All the other formats refer to accession numbers, not sequences.

It is best to keep the default in the various selectors in the boxes on the right side of the form:

- **Gapped Alignments On/Off:** Makes it possible to force BLAST to behave like some more ancient version of this program that did not allow for gaps. To get the same behavior as the NCBI Blast, keep this box checked.
- **BLAST Filter On/Off:** Switches off the low complexity filter. This is to prevent regions where an amino-acid is repeated many times from biasing your search. Having this on is the default, and you should keep it that way unless you have a good reason. (See the section "Controlling the sequence masking," later in this chapter.)
- **Graphic Display On/Off:** Keep this on if you want the same graphic display as the NCBI.

5. **Enter the ID number (or the sequence itself) in the sequence box.**

   We're using **P09405** as our ID number. If you have a sequence, paste it from the Clipboard into the same window.

6. **Click the Run BLAST button.**
7. **Save your results.**

   BLAST outputs can be HUGE: Never print them out (unless you plan to rid your whole department of its stock of paper).

   When saving BLAST results, preserving the graphic display is notoriously difficult. If you want to keep this display, save it separately: Put the mouse pointer over the image, right-click, and then choose Save Picture As from the context menu that appears.

   If you have access to a PDF printer program, print your result into a PDF file; this is the best way to keep a printable electronic record.

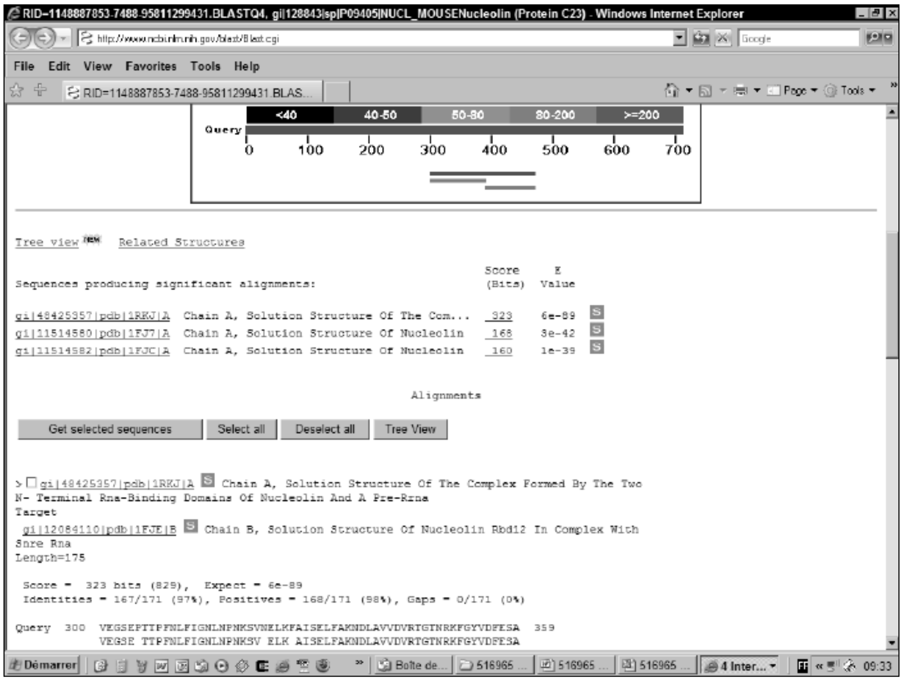
Understanding your BLAST output

All the flavors of BLAST return a similar output. This output is rather complicated and can be confusing, even for experienced users. Figure 7-7 shows you

You'll find four sections in the output of most BLAST servers. These sections always appear in the same order, shown in Figure 7-7, and include

1. **A graphic display:** Shows you where your query is similar to other sequences. Depending on the server you use, this display can change a lot. It may also be absent on some servers.
2. **A hit list:** The name of sequences similar to your query, ranked by similarity.
3. **The alignments:** Every alignment between your query and the reported hits.
4. **The parameters:** A list of the various parameters used for the search.

Each of these elements contains a lot of information. Knowing what matters in each of these displays can be very useful to your research.


RID=1148887853-7488-95811299431.BLASTQ4, gi|128843|sp|P09405|NUCL_MOUSENucleolin (Protein C23) - Windows Internet Explorer
http://www.ncbi.nlm.nih.gov/blast/Blast.cgi
Google
File Edit View Favorites Tools Help
RID=1148887853-7488-95811299431.BLAS...
Page
Tools
<40
40-50
50-80
80-200
>=200
Query
0
100
200
300
400
500
600
700
Tree view NEW
Related Structures
Get selected sequences
Select all
Deselect all
Tree View
Démarrer
Boîte de...
516965 ...
516965 ...
516965 ...
4 Inter...
09:33

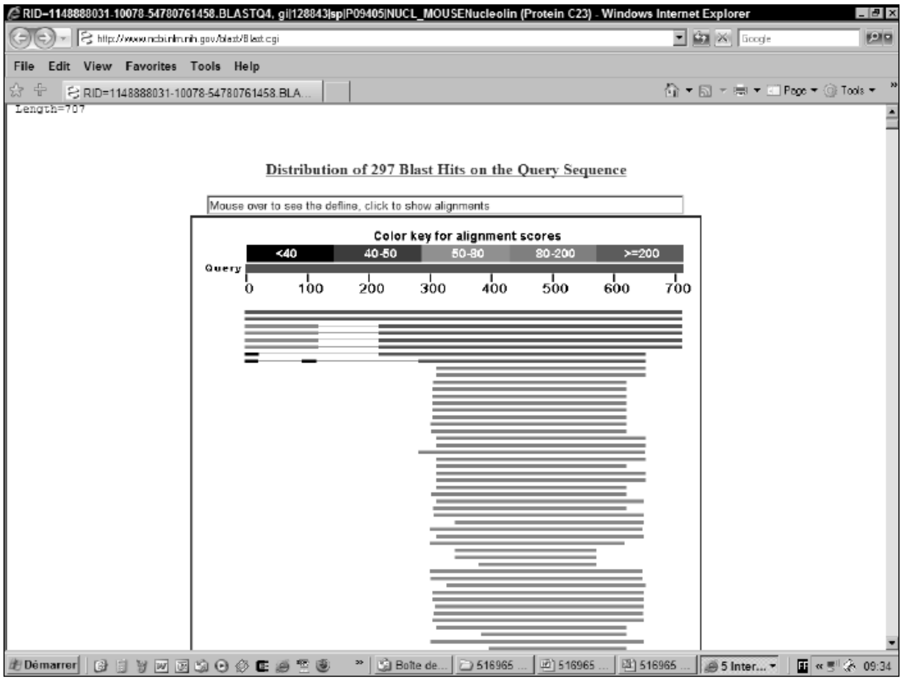
```
                                                                 Score     E
Sequences producing significant alignments:                      (Bits)  Value

gi|48425357|pdb|1RKJ|A  Chain A, Solution Structure Of The Com...   323    6e-89  S
gi|11514580|pdb|1FJ7|A  Chain A, Solution Structure Of Nucleolin    168    3e-42  S
gi|11514582|pdb|1FJC|A  Chain A, Solution Structure Of Nucleolin    160    1e-39  S

                                 Alignments

> □ gi|48425357|pdb|1RKJ|A  S Chain A, Solution Structure Of The Complex Formed By The Two
N- Terminal Rna-Binding Domains Of Nucleolin And A Pre-Rrna
Target
 gi|12084110|pdb|1FJE|B  S Chain B, Solution Structure Of Nucleolin Rbd12 In Complex With
Snre Rna
Length=175

 Score =  323 bits (829),  Expect = 6e-89
 Identities = 167/171 (97%), Positives = 168/171 (98%), Gaps = 0/171 (0%)

Query  300  VEGSEPTTPFNLFIGNLNPNKSVNELKFAISELFAKNDLAVVDVRTGTNRKFGYVDFESA  359
            VEGSE TTPFNLFIGNLNPNKSV ELK AISELFAKNDLAVVDVRTGTNRKFGYVDFESA
```

Figure 7-7: Schematic representation of the main compo nents of a BLAST output (does not include the parameters section at the bottom).

pink bars indicate matches that are a bit less good; green bars indicate matches that are not impressive at all. The blue and the black bars indicate matches that have the worst scores.

Red, pink, and green matches are usually the good ones. Black hits are the bad hits: proteins that have so little in common with the query that their alignment probably means nothing from a biological point of view. (They come from the twilight zone.)

The nice thing about this display is that it can help you see that some matches do not extend over the complete length of your sequence. It is a useful tool to discover domains. For instance, here in Figure 7-8, the top hits are proteins homologous to the nucleolin. Near the bottom, the shorter hits correspond to the domain in the nucleolin that binds RNA. These hits indicate proteins that also contain this RNA binding domain but are otherwise unrelated to the nucleolin.


RID=1148888031-10078-54780761458.BLASTQ4, gi|128843|sp|P09405|NUCL_MOUSENucleolin (Protein C23) - Windows Internet Explorer
http://www.ncbi.nlm.nih.gov/blast/Blast.cgi
Google
File Edit View Favorites Tools Help
RID=1148888031-10078-54780761458.BLA...
Page
Tools
Démarrer
Boîte de...
516965...
516965...
516965...
5 Inter...
09:34

Length=707

Distribution of 297 Blast Hits on the Query Sequence

Mouse over to see the defline, click to show alignments

Color key for alignment scores
<40
40-50
50-80
80-200
>=200
Query
0
100
200
300
400
500
600
700

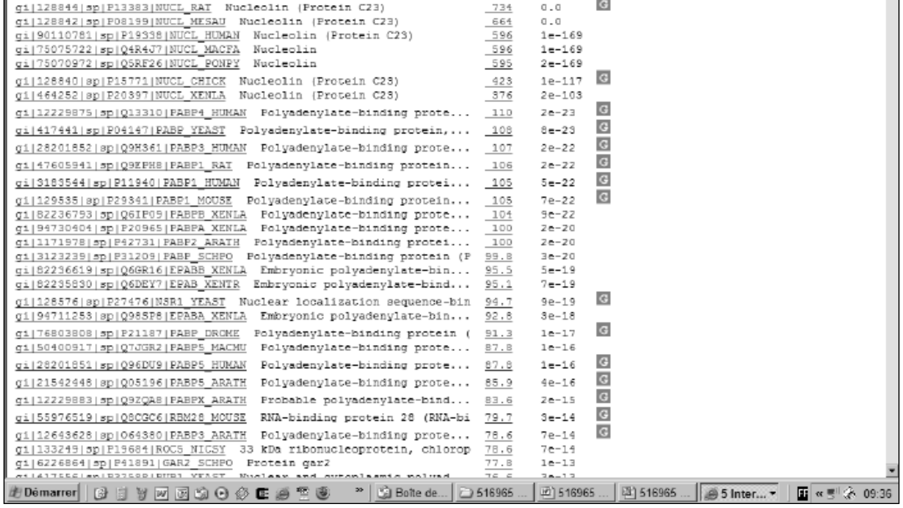
Figure 7-8: The NCBI BLAST graphic display.

When two colored lines are linked by a thin black line, as happens on the third line in Figure 7-8, it means that the same protein matches your query in two separate locations. (Please note that the thin black line is what you see with NCBI; on the EMBNet server, you'll see a dashed black line.) These matches are independent, but they could probably be joined to form a longer match.

The hit list

Most BLAST users consider the hit list (like the one shown in Figure 7-9) their favorite part of the BLAST output. It immediately tells you whether your sequence looks like something that's already in the database — and whether you can trust it as a good hit. Each line contains four important features:

- **The sequence accession number and the name:** This hyperlink takes you to the database entry that contains this sequence. In this entry, you may find very important annotated information describing the sequence. A Swiss-Prot link (|sp| in Figure 7-9) may tempt you with potentially useful information.
- **Description:** A description that comes from the annotation lets you know at a glance whether this finding could be interesting for you. Of course, you never have a *guarantee* that this annotation is correct; we recommend that you carefully check the complete annotation before getting overly excited.
- **The bit score:** A measure of the statistical significance of the alignment. The higher the bit score, the more similar the two sequences. Matches below 50 bits are very unreliable.
- **The E-value (the expectation value):** By estimating the number of times you could have expected such a good match only by chance (given the database), the E-value provides you with the most important measure of statistical significance. (See the nearby sidebar, "E-values, similarity, and homology.") The lower the E-value, the more similar the sequences and the more confidence you can have that this hit is really homologous to your query. For instance, sequences identical to your query have E-values very close to 0. Matches above 0.001 are often close to the twilight zone.
- **Genomic link:** Now that so many complete genomes are available, you probably want to know about the genomic location of potential matches. Whenever such information is available, it is indicated by a purple G. Just click the G and you'll find out everything you need to know about this gene.



```
gi|128844|sp|P13383|NUCL_RAT  Nucleolin (Protein C23)                  734     0.0
gi|128842|sp|P08199|NUCL_MESAU  Nucleolin (Protein C23)                664     0.0
gi|90110781|sp|P19338|NUCL_HUMAN  Nucleolin (Protein C23)              596     1e-169
gi|75075722|sp|Q4R4J7|NUCL_MACFA  Nucleolin                            596     1e-169
gi|75070972|sp|Q5RF26|NUCL_PONPY  Nucleolin                            595     2e-169
gi|128840|sp|P15771|NUCL_CHICK  Nucleolin (Protein C23)                423     1e-117
gi|464252|sp|P20397|NUCL_XENLA  Nucleolin (Protein C23)                376     2e-103
gi|12229875|sp|Q13310|PABP4_HUMAN  Polyadenylate-binding prote...      110     2e-23
gi|417441|sp|P04147|PABP_YEAST  Polyadenylate-binding protein,...      108     8e-23
gi|28201852|sp|Q9H361|PABP3_HUMAN  Polyadenylate-binding prote...      107     2e-22
gi|47605941|sp|Q9EPH8|PABP1_RAT  Polyadenylate-binding protein...      106     2e-22
gi|3183544|sp|P11940|PABP1_HUMAN  Polyadenylate-binding protei...      105     5e-22
gi|129535|sp|P29341|PABP1_MOUSE  Polyadenylate-binding protein...      105     7e-22
gi|82236793|sp|Q6IP09|PABPB_XENLA  Polyadenylate-binding prote...      104     9e-22
gi|94730404|sp|P20965|PABPA_XENLA  Polyadenylate-binding prote...      100     2e-20
gi|1171978|sp|P42731|PABP2_ARATH  Polyadenylate-binding protei...      100     2e-20
gi|3123239|sp|P31209|PABP_SCHPO  Polyadenylate-binding protein (P     99.8     3e-20
gi|82236619|sp|Q6GR16|EPABB_XENLA  Embryonic polyadenylate-bin...     95.5     5e-19
gi|82235830|sp|Q6DEY7|EPAB_XENTR  Embryonic polyadenylate-bind...     95.1     7e-19
gi|128576|sp|P27476|NSR1_YEAST  Nuclear localization sequence-bin     94.7     9e-19
gi|94711253|sp|Q98SP8|EPABA_XENLA  Embryonic polyadenylate-bin...     92.8     3e-18
gi|76803808|sp|P21187|PABP_DROME  Polyadenylate-binding protein (     91.3     1e-17
gi|50400917|sp|Q7JGR2|PABP5_MACMU  Polyadenylate-binding prote...     87.8     1e-16
gi|28201851|sp|Q96DU9|PABP5_HUMAN  Polyadenylate-binding prote...     87.8     1e-16
gi|21542448|sp|Q05196|PABP5_ARATH  Polyadenylate-binding prote...     85.9     4e-16
gi|12229883|sp|Q9ZQA8|PABPX_ARATH  Probable polyadenylate-bind...     83.6     2e-15
gi|55976519|sp|Q8CGC6|RBM28_MOUSE  RNA-binding protein 28 (RNA-bi     79.7     3e-14
gi|12643628|sp|O64380|PABP3_ARATH  Polyadenylate-binding prote...     78.6     7e-14
gi|133249|sp|P19684|ROC5_NICSY  33 kDa ribonucleoprotein, chlorop     78.6     7e-14
gi|6226864|sp|P41891|GAR2_SCHPO  Protein gar2                         77.8     1e-13
gi|417556|sp|P32588|PUB1_YEAST  Nuclear and cytoplasmic polyad...     76.6     3e-13
```

Figure 7-9: The NCBI BLAST hit list.

The alignments

No matter what we say about E-values and statistical significance, a number is only a number. When it comes to telling the real story, biologists only trust alignments. Biologists are convinced that alignments cannot lie, and this is mostly true — if you know how to look at them.

BLAST displays the alignments for a BLAST search just below the hit list, as shown in Figure 7-10. In every BLAST alignment, you can find the following features.

- **The name(s):** The first item is the name. Sometimes there is more than one sequence name; this happens whenever several sequences align identically to those in the query. For instance, if the database contains several sequences that are identical to the ones in the query, all these matches will be reported as a single hit in the alignment section. This may even happen with so-called Non-Redundant databases!
- **The percentage of identity:** This gives you a concrete substitute for the E-value. Remember the magic number: An identity of more than 25 percent is good news. (The *identity* is the number of identical residues divided by the number of matched residues — gaps are simply ignored.) The `Positives` field gives you a measure of the fraction of residues that are either identical or similar — represented with a + on the actual alignment. The `Gaps` field shows residues that were not aligned.

evolved from a common ancestor and have the same overall 3-D structure. Biologists call these sequences *homologues.* In practice, homologue sequences often have similar biochemical functions.

If you are studying a protein, the most desirable object in the universe is a very well-characterized protein sequence that's clearly homologous to your protein. People search databases in hopes of finding this special sequence.

That's all very fine, but how do you show that your two sequences are homologous? Think of homologous sequences as relatives in a family. We all know that relatives tend to look alike, but we also know that two persons with the same eye color aren't necessarily siblings. On the other hand, if they have the same type of hair, the same facial features, and so on, we can be tempted to conclude that they are true relatives. It works the same way with sequences.

How similar must sequences be in order to be considered homologous? The answer is clear: More than *25 percent* of the amino acids present for proteins — and more than *70 percent* of the nucleotides present for DNA — must be similar. Above this limit, you can be almost sure that two proteins have the same structure and the same common ancestor. Below that limit lies the twilight zone — that spooky identity range where nobody can really be sure whether the observed similarity means anything.

Warning: Be careful! The *25-percent* and *70-percent* limits only work for sequences that contain more than 100 amino acids or nucleotides. You might frequently get near-perfect identity between short segments (10 residues, for instance) of totally unrelated proteins or DNA sequences.

Everybody loves percent identities because they are so easy to spot visually. Unfortunately, similar — but not identical — amino acids are aligned together. Moreover, how do you tell the difference between 60 matched residues spread over a 100-residue segment, and 120 matches spread over a 200-residue segment? The longest is probably more meaningful, but the percent identity says nothing about this.

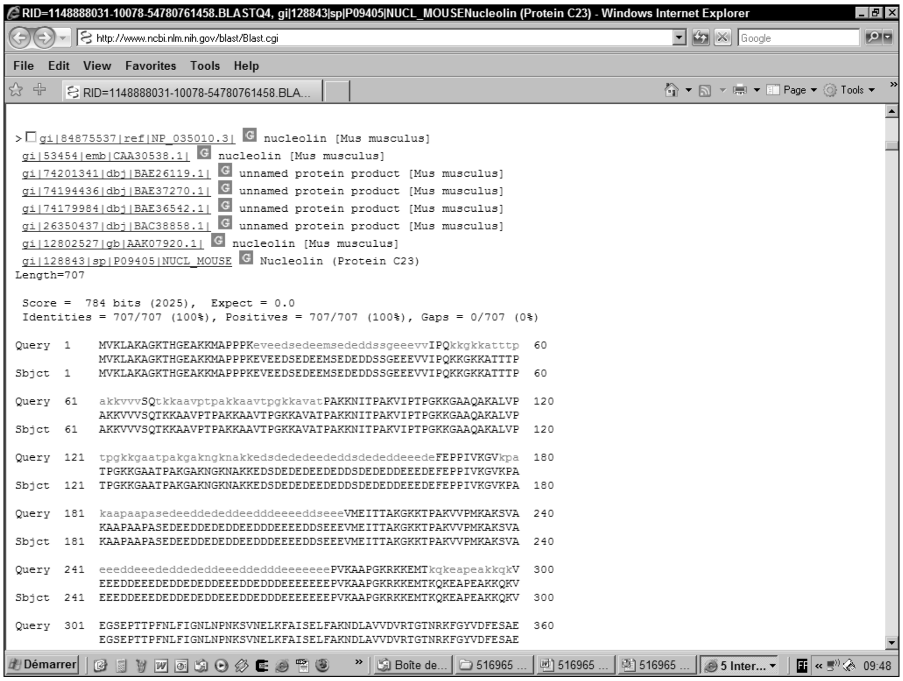
Gurus invented E-values so we'd have a criterion more objective than percentage-of-similarity. E-values (short for *expectation values*) are a powerful tool for comparing pairwise alignments with different similarities and different lengths. They also help you decide how much you can trust your conclusion on homology. With E-values, you know when you can get excited or when you should wait and see.

The E-value has a very concrete meaning: It is the number of times your database match may have occurred just by chance. We consider a match that's very unlikely to occur just by chance to be a very good match; that's why results associated with the *lowest* E-values are the best. We say they're the most *significant* because we know we can trust them enough to infer homology.

In theory, alignments associated with E-values lower than one should all be trusted. In practice, this is not true because BLAST uses an approximate formula for computing the E-values and strongly underestimates them. In the sequence world, a similarity with an E-value above 10-^{4} (0.0001) is not necessarily interesting. If you want to be certain of the homology, your E-value must be lower than 10-^{4}.

BLAST isn't the only program that uses E-values. You may come across them almost any time you compare two sequences — even if you use programs that compare sequences and domains. The principle is always the same.

- **The bottom sequence:** The “hit” — here referred to as the “subject.”
- **The line in between the sequences:** Between the two sequences is a line that contains a (+) sign for similar amino acids, a letter for identical residues, or a space for mismatches.
- **The XXXXX regions:** In your query, BLAST inserts Xs automatically to mask the regions that contain many identical residues. Gurus call these regions the *low-complexity segments.* They can cause trouble in the search; the Xs tell BLAST to ignore the corresponding segments. The masking occurs only in the query sequence.
- **The numbers:** Numbers to the right side of the sequences indicate the coordinates of the match on your query sequence and on the hit sequence. BLAST makes local alignments that may contain only a portion of the query and the hit.


RID=1148888031-10078-54780761458.BLASTQ4, gi|128843|sp|P09405|NUCL_MOUSENucleolin (Protein C23) - Windows Internet Explorer
http://www.ncbi.nlm.nih.gov/blast/Blast.cgi
Google
File
Edit
View
Favorites
Tools
Help
RID=1148888031-10078-54780761458.BLA...
Page
Tools
Démarrer
Boîte de...
516965 ...
516965 ...
516965 ...
5 Inter...
FR
09:48

```
> □gi|84875537|ref|NP_035010.3| G nucleolin [Mus musculus]
 gi|53454|emb|CAA30538.1| G nucleolin [Mus musculus]
 gi|74201341|dbj|BAE26119.1| G unnamed protein product [Mus musculus]
 gi|74194436|dbj|BAE37270.1| G unnamed protein product [Mus musculus]
 gi|74179984|dbj|BAE36542.1| G unnamed protein product [Mus musculus]
 gi|26350437|dbj|BAC38858.1| G unnamed protein product [Mus musculus]
 gi|12802527|gb|AAK07920.1| G nucleolin [Mus musculus]
 gi|128843|sp|P09405|NUCL_MOUSE G Nucleolin (Protein C23)
Length=707

 Score =  784 bits (2025),  Expect = 0.0
 Identities = 707/707 (100%), Positives = 707/707 (100%), Gaps = 0/707 (0%)

Query  1    MVKLAKAGKTHGEAKKMAPPPKeveedsedeemsededdssgeeevvIPQkkgkkatttp  60
            MVKLAKAGKTHGEAKKMAPPPKEVEEDSEDEEMSEDEDDSSGEEEVVIPQKKGKKATTTP
Sbjct  1    MVKLAKAGKTHGEAKKMAPPPKEVEEDSEDEEMSEDEDDSSGEEEVVIPQKKGKKATTTP  60

Query  61   akkvvvSQtkkaavptpakkaavtpgkkavatPAKKNITPAKVIPTPGKKGAAQAKALVP  120
            AKKVVVSQTKKAAVPTPAKKAAVTPGKKAVATPAKKNITPAKVIPTPGKKGAAQAKALVP
Sbjct  61   AKKVVVSQTKKAAVPTPAKKAAVTPGKKAVATPAKKNITPAKVIPTPGKKGAAQAKALVP  120

Query  121  tpgkkgaatpakgakngknakkedsdededeededdsdededdeeedeFEPPIVKGVkpa  180
            TPGKKGAATPAKGAKNGKNAKKEDSDEDEDEEDEDDSDEDEDDEEEDEFEPPIVKGVKPA
Sbjct  121  TPGKKGAATPAKGAKNGKNAKKEDSDEDEDEEDEDDSDEDEDDEEEDEFEPPIVKGVKPA  180

Query  181  kaapaapasedeeddededdeedddeeeeddseeeVMEITTAKGKKTPAKVVPMKAKSVA  240
            KAAPAAPASEDEEDDEDEDDEEDDDEEEEDDSEEEVMEITTAKGKKTPAKVVPMKAKSVA
Sbjct  181  KAAPAAPASEDEEDDEDEDDEEDDDEEEEDDSEEEVMEITTAKGKKTPAKVVPMKAKSVA  240

Query  241  eeeddeeededdededdeeeddedddeeeeeeePVKAAPGKRKKEMTkqkeapeakkqkV  300
            EEEDDEEEDEDDEDEDDEEEDDEDDDEEEEEEEPVKAAPGKRKKEMTKQKEAPEAKKQKV
Sbjct  241  EEEDDEEEDEDDEDEDDEEEDDEDDDEEEEEEEPVKAAPGKRKKEMTKQKEAPEAKKQKV  300

Query  301  EGSEPTTPFNLFIGNLNPNKSVNELKFAISELFAKNDLAVVDVRTGTNRKFGYVDFESAE  360
            EGSEPTTPFNLFIGNLNPNKSVNELKFAISELFAKNDLAVVDVRTGTNRKFGYVDFESAE
```

Figure 7-10: Pairwise alignments reported by BLAST.

happens when the query and the hit sequence could be aligned in several locations. On the graphic display, a thin black line (NCBI) or a dashed line (EMBnet) connects these independent alignments.

The parameters

You don't need to worry much about the meaning of the information that comes at the bottom of the result page. If you've changed the default parameters of BLAST, this part keeps track of it for you, making sure that you can reproduce this search if you need it.

Nobody can guarantee that you can reproduce a result exactly, even if you use the same BLAST server. The upgrade of any of the components in the server can modify the results of the search. Components that may be upgraded include

- The databases
- The BLAST program itself
- The default parameters of the server

In real life, you have no control over these parameters. They always end up changing with time. At least the information returned at the bottom of the search may offer a clue about what *could* be going wrong.

BLASTing DNA sequences

If you think that BLASTing a DNA sequence is like BLASTing a protein sequence, you're right — and you're not right at the same time. While it's true that the principle is the same and that in practice BLASTing DNA requires operations similar to BLASTing proteins, BLASTing DNA does not always work so well. It is faster and more accurate to BLAST proteins (blastp) rather than nucleotides.

If you know the reading frame in your sequence, you're better off translating the sequence yourself and BLASTing with a protein sequence (see the corresponding section, "BLASTing protein sequences," earlier in this chapter). If you don't know the reading frame, then you must choose one of the following BLAST programs (see Table 7-2) that deal with nucleotides.

tblastx	TDNA	TDNA	Protein discovery and ESTs
blastx	TDNA	Protein	Analysis of the query DNA sequence

T is for *translated.* It means that you hand over a DNA sequence to BLAST, and BLAST translates this sequence into its six frames (three on one strand and three on the other one). Where there is a Stop codon, BLAST replaces it with an X. The six-frame translation means you needn't worry about the DNA strand you give to BLAST. The program tries each possible frame for you anyway.

Choosing the right flavor of BLAST for your DNA sequence

Although the list of programs in Table 7-2 with all the weird acronyms can look complicated, don't let it intimidate you — and remember that nothing ever breaks in the cyberlab! Just ask yourself the questions in Table 7-3, in the order listed, to determine which is the best program to use.

Table 7-3 Choosing the Right Flavor of BLAST for DNA

Question	*Answer*
Am I interested in non-coding DNA?	Yes: Use blastn. Never forget that blastn is only for closely related DNA sequences (more than 70 percent identical).
Do I want to discover new proteins?	Yes: Use tblastx.
Do I want to discover proteins encoded in my query DNA sequence?	Yes: Use blastx.
Am I unsure of the quality of my DNA?	Yes: Use blastx if you suspect your DNA sequence is the coding for a protein but that it may contain sequencing errors.

TIP

Blastx can correct sequencing errors for you. If you aren't sure of your protein sequence, blastx may reveal its similarity to a better-sequenced piece of DNA. If you find that your sequence contains an unexpected frameshift, don't get immediately excited. Sequencing errors cause most of the observed frame shifts — but double-check in the lab to be sure.

- **Pick the right database:** Choose a database that's compatible with the BLAST program you want to use. Unless you use blastx, it must be a DNA sequence database. Most BLAST servers do not check to make sure that you did this properly and, consequently, let you wait forever for results that never come.
- **Restrict your search:** Database searches on DNA are slower. When possible, restrict your search to the subset of the database that you're interested in. For instance, if you're interested in the Drosophila (fruit fly), search only the Drosophila genome.
- **Shop around:** Don't hesitate to shop around to find a BLAST server that contains the database you're interested in.
- **Use filtering:** Genomic sequences are full of repetitions. Be sure to filter out at least some of them.

The BLAST way of doing things

People who do bioinformatic analysis know *there is almost nothing you can't do with BLAST!* Whatever biological question you're asking, chances are that BLAST can give you some ideas before you start using more complicated programs or spend time designing costly experiments. In Table 7-4, we give you four examples. For each of these applications, there is an exhaustive and complicated *good* solution — and then there's the BLAST way: Quick and efficient, it can be enough in most cases. (Of course, if you have too much time on your hands, you can always go for the hard way.)

Table 7-4 Asking Biological Questions with BLAST

What You Need to Do	*The BLAST Way*	*The Complicated Alternative*
Finding genes in a genome	Cut your genome sequence into little (2-to-5-kilobyte) overlapping sequences. Use blastx to BLAST each piece of genome against NR (the Non Redundant protein database). This works better if you have no introns (bacteria).	Run gene-prediction software, sequence mRNAs

	Swiss-Prot. If you get a good hit (more than 25 percent identity) over the complete length of the protein, you've solved your problem and you know that your protein has the same function as the Swiss-Prot protein.	
Predicting a protein 3-D structure	Use blastp to BLAST your protein against PDB (the database of protein structure). If you get a good hit (an identity of more than 25 percent), you know that your protein and this good hit have a similar 3-D structure.	Conduct homology modeling, X-ray, or NMR analysis of your protein
Finding protein family members	Use blastp (or its more powerful cousin PSI-BLAST) and run it against NR (the non-redundant protein family). After you have all the members of the family, you can make a multiple sequence alignment (see Chapter 9) and draw a phylogenetic tree.	Clone new family members using PCR techniques

Controlling BLAST: Choosing the Right Parameters

As they say, *power is nothing without control.* This saying summarizes fairly well how you can get the best out of BLAST: by controlling it. In this section, we show you the main parameters in BLAST, what they do, and how you can change them to suit your needs.

you don't get anything meaningful with them, don't expect any miracles if you change them. Table 7-5 lists the main reasons why you may want to change the default parameters, along with the parameters you may change. These reasons are valid for DNA and protein sequences alike.

WARNING!

Most of the BLAST servers use slightly different ways of specifying parameters. In this section, we have chosen to focus on the NCBI server. You will find that it is generally easy to adapt what we are saying here to your favorite server, although sometimes the parameters can feel a bit hidden. Whenever you want to customize your search, look for the advanced mode of the server you are using; that often offers more possibilities.

Table 7-5 Some Reasons to Change BLAST Default Parameters

Reason	***Parameters to Change***
The sequence you're interested in contains many identical residues; it has a biased composition.	Sequence filter (automatic masking)
BLAST doesn't report any results.	The substitution matrix or the gap penalties
Your match has a borderline E-value.	The substitution matrix or the gap penalties to check the match's robustness
BLAST reports too many matches.	The database you're searching OR filter the reported entries by keyword OR increase the number of reported matches OR increase Expect (the E-value threshold) OR reject sequences too similar to the query (those with very low E-values)

Controlling the sequence masking

When BLAST searches databases, it makes an important assumption: BLAST assumes that all your sequences are *average* sequences. For instance, if you're searching protein sequences, BLAST supposes that the average protein composition of any protein is the same as the average composition of the whole database. In practice, this assumption is too far from reality to work all the time.

of identical amino acids it contains. Unfortunately, there is a good chance that these two proline-rich domains are not related at all. In fact, these one-amino-acid-rich domains are notorious for fooling BLAST.

To avoid this problem, BLAST filters out low-complexity regions when analyzing proteins. To do that, it replaces those regions in the sequence with Xs. If you're specifically interested in the low-complexity regions and don't want these regions filtered out of your search, you must deselect the corresponding Low Complexity check box next to Choose Filter in the Options for Advanced Blasting section of the blastp search page, as shown in Figure 7-11.

Imagine that you have just cloned and sequenced a protein. In the sequence of this protein, you find a stretch of 10 Prolines: **PPPPPPPPPP**. At this point, you're probably wondering whether this is a sequencing error or a mistake of some kind. You would feel much more confident if you had a valid protein sequence that contains the same stretch of amino acids.

To find this protein, write your own sequence **PPPPPPPPPPPPPPPPPP** and give it to blastp as a query. Do not forget to deselect the Low Complexity check box.

Some domains are very common in protein sequences, such as Zn Fingers in tandems or Fibronectines domains. If your protein contains this type of domain, BLAST reports many matches with proteins that contain the same domains but are otherwise unrelated. To make your search more interesting, filtering out these domains is a good idea. Here's how:

1. **Use CD search, InterProScan, or Pfscan to find domains in your protein.**

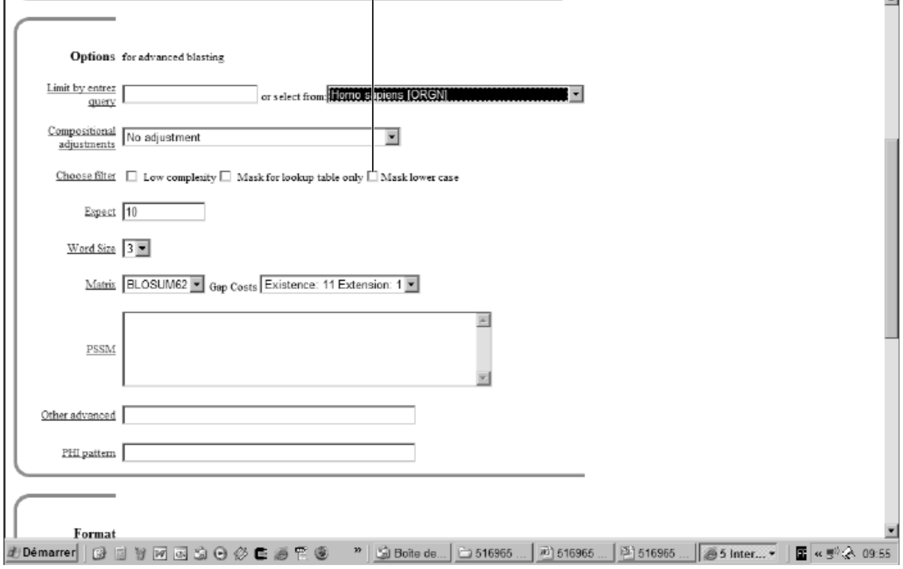
   See Chapter 6 for more information about using these searches.

2. **Read the domain documentation to find out how widespread the domains are that you found.**

3. **Replace the sequences of less-informative domains with Xs (or rewrite them in lowercase) and then select the Mask Lower check box next to Choose Filter in the BLAST form. (See Figure 7-11.)**

4. **Run a standard blastp as shown at the beginning of this chapter.**

   When your results are returned, interpret them as shown in the "Understanding your BLAST output" section, earlier in this chapter.



Options for advanced blasting

Limit by entrez query: or select from: Homo sapiens [ORGN]

Compositional adjustments: No adjustment

Choose filter: □ Low complexity □ Mask for lookup table only □ Mask lower case

Expect: 10

Word Size: 3

Matrix: BLOSUM62 Gap Costs: Existence: 11 Extension: 1

PSSM:

Other advanced:

PHI pattern:

Format

Figure 7-11: The advanced parameters of blastp at NCBI.

Masking DNA sequences

If you think it's a problem keeping your protein-sequence results significant, things get even worse when you're dealing with DNA sequences. DNA is full of sequences that repeat many times, and yet such similarities don't actually mean very much when you come right down to it. Most genomes contain many such repeats — especially the human genome, which is 60 percent repeats! If you want to avoid having to wade through that many repeats, select the Human Repeats check box that appears in the blastn page, as shown in Figure 7-12. Selecting this box causes the human repeats to be filtered out from your sequences.

TIP

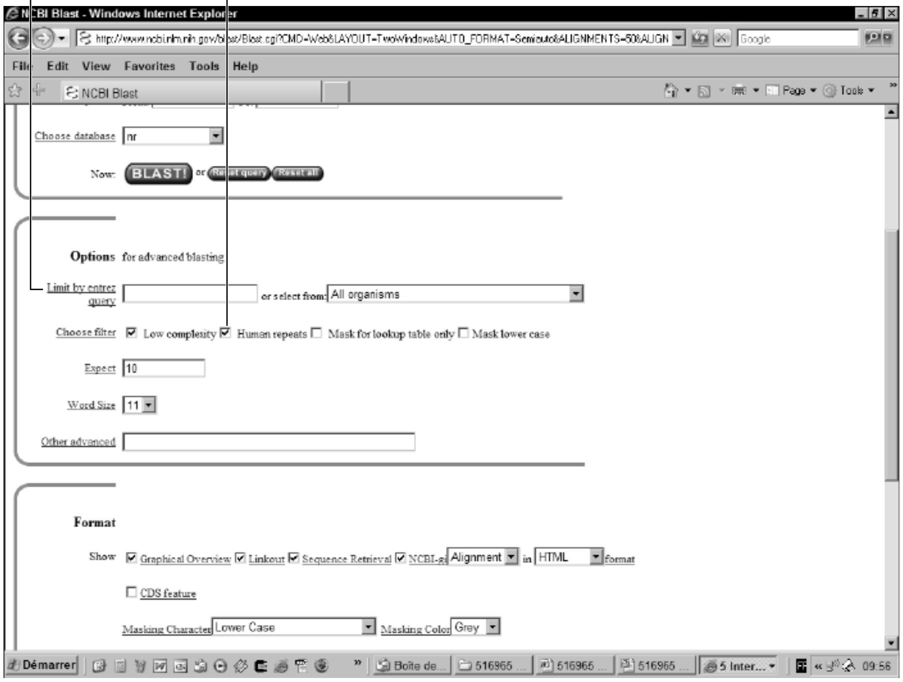
Large-scale genome sequencing is not always as smooth as the gurus want us to believe. For instance, in many cases, the final genome ends up containing bits and pieces of the organism used to clone it (which is why, when you're looking at a human genome, you may find bits of yeast and *E. coli*). This simply means that before you get very excited about a gene you've just found in the human genome (or in any genome), you may want to make sure that this gene is not a contamination brought in by the organism used for the cloning. Imagine

parameters

Figure 7-11 shows you all the parameters you can change on the NCBI BLAST server. These include the Expect value, the Word Size, and the means of filtering. Among these parameters, two important ones have to do with the way BLAST makes the alignments; these are the *gap penalties* (named *gap cost* on the NCBI BLAST server page) and the *substitution matrix (*named simply *matrix* on the NCBI BLAST server page). If you change either of these parameters, BLAST may report different hits. Hits that are likeliest to change from one run to the next are the hits that were borderline (those with an E-value higher than 0.0001).

Limit by Entrez Query option

Human Repeats option



Choose database: nr

Options for advanced blasting

Limit by entrez query: or select from: All organisms

Choose filter: ☑ Low complexity ☑ Human repeats □ Mask for lookup table only □ Mask lower case

Expect: 10

Word Size: 11

Other advanced:

Format

Show: ☑ Graphical Overview ☑ Linkout ☑ Sequence Retrieval ☑ NCBI-gi Alignment in HTML format

□ CDS feature

Masking Character: Lower Case Masking Color: Grey

NCBI Blast - Windows Internet Explorer
http://www.ncbi.nlm.nih.gov/blast/Blast.cgi?CMD=Web&LAYOUT=TwoWindows&AUTO_FORMAT=Semiauto&ALIGNMENTS=50&ALIGN
File Edit View Favorites Tools Help
NCBI Blast
Now: BLAST! or Reset query Reset all
Démarrer
Boite de...
516965 ...
516965 ...
516965 ...
5 Inter...
09:56

Figure 7-12: Filtering possibilities for a DNA sequence while using blastn.

faster and less sensitive; short words do the opposite.

The best reason to play with these parameters is to check the robustness of a borderline hit. If this match does not go away when you change the substitution matrix or the gap penalties, then it has better chances of being a biologically meaningful match.

Controlling the BLAST output

If the subject of your query belongs to a large protein family, the BLAST output may give you trouble because the databases contain too many sequences nearly identical to yours.

Sometimes this wealth of homologous sequences can prevent you from seeing a homologous sequence that's less closely related but still associated with experimental information. If this sort of thing happens to you, refer to Table 7-5 — where the last entry gives you five solutions for correcting this problem. In the following sections, we show you how to implement those solutions.

Choosing the right database

Choose a database that is suited to your needs. If BLAST reports too many hits, you may find a way around this quandary by searching Swiss-Prot rather than NR (Swiss-Prot is 100 times smaller than NR), by searching only one genome, or by only searching the PDB if you are mostly interested in structural similarities.

Limit by Entrez query

Use the Limit Results by Entrez Query box (Refer to Figure 7-12) or its associated pull-down menu if you get too many hits when you're interested in only one type of organism.

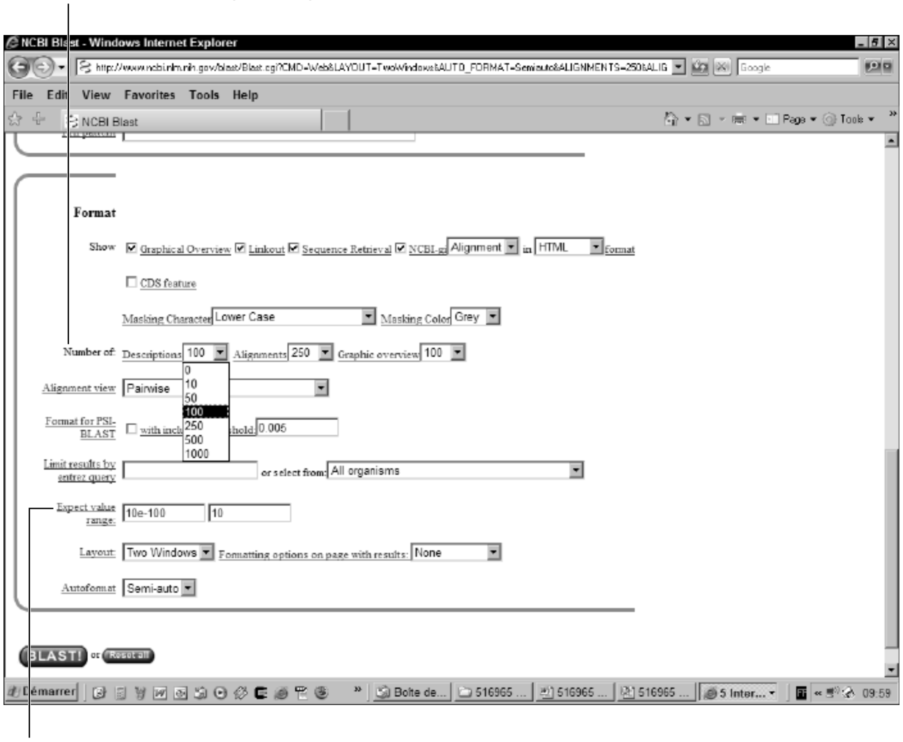
You can use this box to combine keywords with the two Boolean operators AND and NOT. For instance, if you want BLAST to report proteases only — and to ignore proteases from the HIV virus — type **"protease NOT hiv1[Organism]"** (don't forget the quotation marks). If you want to find more about this, see the online documentation on Entrez queries at

E-value higher than what you indicate in the corresponding box. A low cutoff value for this parameter forces BLAST to report only good hits.

You can filter on both side of the range and prevent the report of semi-identical sequences. For instance, in Figure 7-13, we requested that the best sequences (E-value lower than 10e–100) not be shown.

Be aware that if your search yields more than 100 hits, you also need to increase the number of reported Descriptions in the Format section, as shown in Figure 7-13.

Number of Descriptions option


NCBI Blast - Windows Internet Explorer
http://www.ncbi.nlm.nih.gov/blast/Blast.cgi?CMD=Web&LAYOUT=TwoWindows&AUTO_FORMAT=Semiauto&ALIGNMENTS=250&ALIG
File Edit View Favorites Tools Help
NCBI Blast
BLAST! or Reset all
Démarrer
Boîte de...
516965 ...
516965 ...
516965 ...
5 Inter...
09:59

Format

Show ☑ Graphical Overview ☑ Linkout ☑ Sequence Retrieval ☑ NCBI-g Alignment in HTML format

☐ CDS feature

Masking Character Lower Case Masking Color Grey

Number of: Descriptions 100 Alignments 250 Graphic overview 100

0
10
50
100
250
500
1000

Alignment view Pairwise

Format for PSI-BLAST ☐ with incl...hold: 0.005

Limit results by entrez query or select from: All organisms

Expect value range: 10e-100 10

Layout: Two Windows Formatting options on page with results: None

Autoformat Semi-auto

Expect Value Range option

Figure 7-13: Reformatting options of BLAST.

closely related sequences. How about finding the other distant cousins, the ones that your query sequence can't recognize?

PSI-BLAST does just that; first it looks for sequences that are closely related to yours — and then, gradually and carefully, it extends the circle of friends to include sequences that didn't look so interesting at first but happen to be related in the end.

Each time PSI-BLAST makes a new attempt to recruit new members of the family, it does so by using the most common properties among the members already recruited. Each new round is an *iteration.*

PSI-BLASTing protein sequences

In the following example, we use PSI-BLAST to ask whether leghemoglobin, a protein that sequesters oxygen in plants, is related to human hemoglobin.

1. **Point your browser to the main NCBI BLAST page at**

   ```
   www.ncbi.nlm.nih.gov/BLAST
   ```

2. **Click the** PSI- and PHI-BLAST **link (the second link under** `protein BLAST`**).**

   The PSI-BLAST page appears, as shown in Figure 7-14.

3. **Enter your sequence accession number or paste the complete sequence in the window.**

   The procedure to give a sequence to PSI-BLAST is exactly the same as the procedure blastp uses. (Refer to Figures 7-2 and 7-3.)

   - If your sequence is already in a database, you can type in its ID. For this example, we use the human hemoglobin, a protein from Swiss-Prot whose accession number is P01922.
   - If your sequence isn't in a database, you must provide it in the FASTA format.

4. **Choose PDB from the Choose Database pull-down menu (refer to Figure 7-14).**

   PDB is the Protein Database, a database of proteins with a known 3-D structure. We chose it here because it is a small database that lends itself well to this example.

instead of PDB. NR may contain intermediates between your sequence and another PDB sequence that PSI-BLAST may catch after a few iterations.

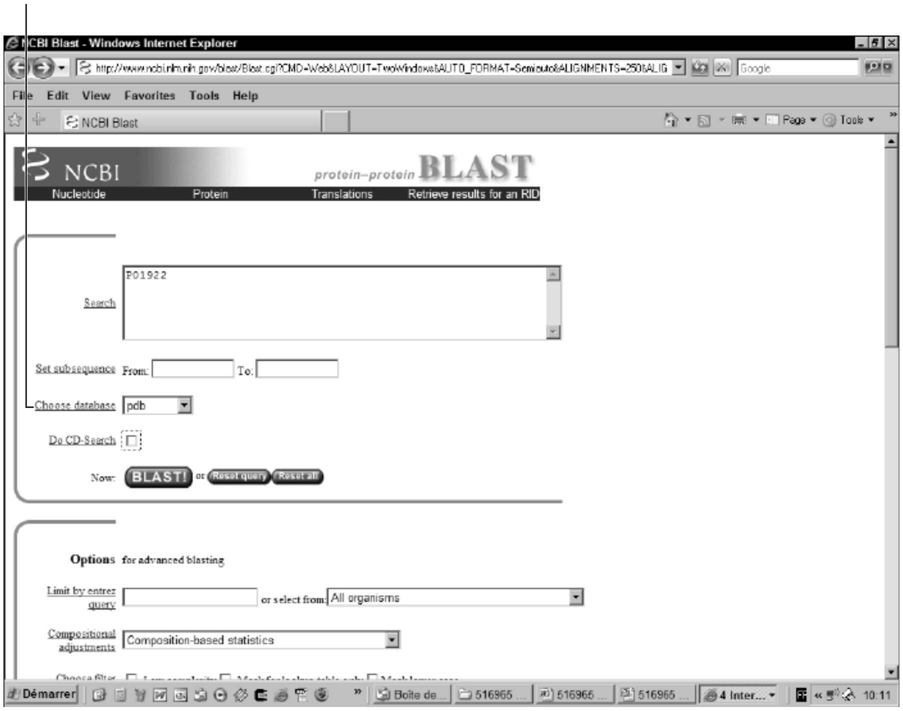
5. **Click the BLAST! button.**

   The time it takes can be longer than what it says on-screen. Be patient!

   An intermediate page (titled Reformatting BLAST) appears, containing a Format! button.

6. **Click this Format! button.**

   A new page appears in a new window titled Results of BLAST. This is where your results are displayed when ready.



Choose Database menu

Search: P01922

Set subsequence From: To:

Choose database: pdb

Do CD-Search □

Options for advanced blasting

Limit by entrez query: or select from: All organisms

Compositional adjustments: Composition-based statistics

NCBI Blast - Windows Internet Explorer
http://www.ncbi.nlm.nih.gov/blast/Blast.cgi?CMD=Web&LAYOUT=TwoWindows&AUTO_FORMAT=Semiauto&ALIGNMENTS=250&ALIG
Google
File Edit View Favorites Tools Help
NCBI Blast
Page
Tools
NCBI
protein–protein BLAST
Nucleotide
Protein
Translations
Retrieve results for an RID
Now:
BLAST!
or
Reset query
Reset all
Choose filter
Démarrer
Boite de...
516965 ...
516965 ...
516965 ...
4 Inter...
10:11

Figure 7-14: Giving a sequence to PSI-BLAST.

a few myoglobins, as we might have expected.

Note in Figure 7-15 that the hit list in PSI-BLAST is different from the one we see in BLAST proper. It contains three new features associated with each sequence:

- **A check box:** If this box is selected, PSI-BLAST is going to use the corresponding sequence to derive the position-specific matrix for the next iteration of PSI-BLAST.
- **The New symbol:** It shows you sequences that PSI-BLAST reports as hits for the first time.
- **The green pill:** It shows you sequences that PSI-BLAST has already used to obtain the current result.

If the description makes you think one of the sequences is not related to your initial query, don’t hesitate to uncheck it. That way, the unrelated sequence won’t be used in the next iteration.

8. **Click the Run PSI-BLAST Iteration 2 button (near the top of the page).**

   The Reformatting BLAST window pops up.

9. **Click the Format! button on the Reformatting BLAST window.**

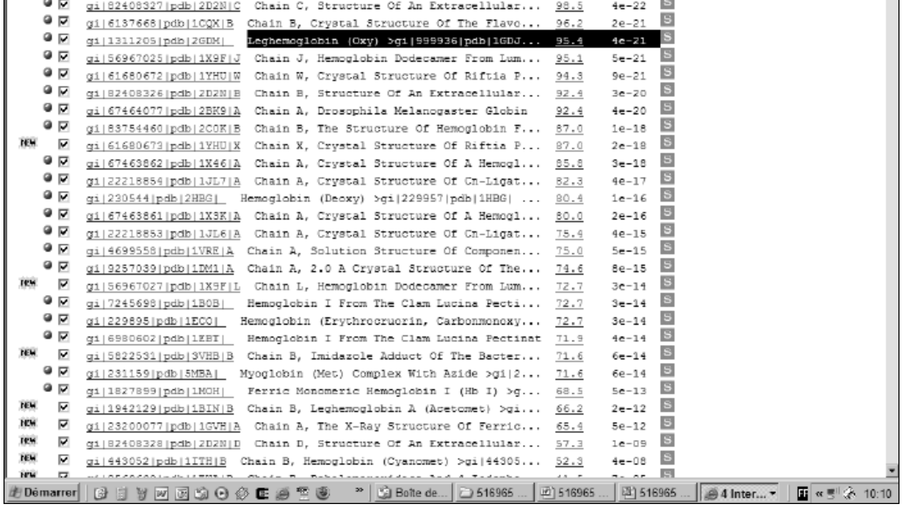
   The results appear in the Result of BLAST window.

10. **Continue repeating Steps 9 and 10.**

    Leghemoglobin appears at the second iteration, but it doesn’t have a very good E-value. However, after the third iteration (Figure 7-15), its E-value improves noticeably, suggesting that it is a genuine relative of the hemoglobin family.

Avoiding mistakes when running PSI-BLAST

In the leghemoglobin example (in the preceding section), we keep going from one iteration to the next because the annotated sequences that every iteration adds are obviously hemoglobin-related. That’s a sign we’re going in the right direction.



```
    ● ☑ gi|82408327|pdb|2D2N|C  Chain C, Structure Of An Extracellular...  98.5  4e-22
    ● ☑ gi|6137668|pdb|1CQX|B  Chain B, Crystal Structure Of The Flavo...  96.2  2e-21
    ● ☑ gi|1311205|pdb|2GDM|   Leghemoglobin (Oxy) >gi|999936|pdb|1GDJ...  95.4  4e-21
    ● ☑ gi|56967025|pdb|1X9F|J  Chain J, Hemoglobin Dodecamer From Lum...  95.1  5e-21
    ● ☑ gi|61680672|pdb|1YHU|W  Chain W, Crystal Structure Of Riftia P...  94.3  9e-21
    ● ☑ gi|82408326|pdb|2D2N|B  Chain B, Structure Of An Extracellular...  92.4  3e-20
    ● ☑ gi|67464077|pdb|2BK9|A  Chain A, Drosophila Melanogaster Globin    92.4  4e-20
    ● ☑ gi|83754460|pdb|2C0K|B  Chain B, The Structure Of Hemoglobin F...  87.0  1e-18
NEW   ☑ gi|61680673|pdb|1YHU|X  Chain X, Crystal Structure Of Riftia P...  87.0  2e-18
    ● ☑ gi|67463862|pdb|1X46|A  Chain A, Crystal Structure Of A Hemogl...  85.8  3e-18
    ● ☑ gi|22218854|pdb|1JL7|A  Chain A, Crystal Structure Of Cn-Ligat...  82.3  4e-17
    ● ☑ gi|230544|pdb|2HBG|   Hemoglobin (Deoxy) >gi|229957|pdb|1HBG| ...  80.4  1e-16
    ● ☑ gi|67463861|pdb|1X3K|A  Chain A, Crystal Structure Of A Hemogl...  80.0  2e-16
    ● ☑ gi|22218853|pdb|1JL6|A  Chain A, Crystal Structure Of Cn-Ligat...  75.4  4e-15
    ● ☑ gi|4699558|pdb|1VRE|A  Chain A, Solution Structure Of Componen...  75.0  5e-15
    ● ☑ gi|9257039|pdb|1DM1|A  Chain A, 2.0 A Crystal Structure Of The...  74.6  8e-15
NEW   ☑ gi|56967027|pdb|1X9F|L  Chain L, Hemoglobin Dodecamer From Lum...  72.7  3e-14
    ● ☑ gi|7245698|pdb|1B0B|   Hemoglobin I From The Clam Lucina Pecti...  72.7  3e-14
    ● ☑ gi|229895|pdb|1ECO|   Hemoglobin (Erythrocruorin, Carbonmonoxy...  72.7  3e-14
    ● ☑ gi|6980602|pdb|1EBT|   Hemoglobin I From The Clam Lucina Pectinat  71.9  4e-14
NEW   ☑ gi|5822531|pdb|3VHB|B  Chain B, Imidazole Adduct Of The Bacter...  71.6  6e-14
    ● ☑ gi|231159|pdb|5MBA|   Myoglobin (Met) Complex With Azide >gi|2...  71.6  6e-14
    ● ☑ gi|1827899|pdb|1MOH|   Ferric Monomeric Hemoglobin I (Hb I) >g...  68.5  5e-13
NEW   ☑ gi|1942129|pdb|1BIN|B  Chain B, Leghemoglobin A (Acetomet) >gi...  66.2  2e-12
NEW   ☑ gi|23200077|pdb|1GVH|A  Chain A, The X-Ray Structure Of Ferric...  65.4  5e-12
NEW   ☑ gi|82408328|pdb|2D2N|D  Chain D, Structure Of An Extracellular...  57.3  1e-09
NEW   ☑ gi|443052|pdb|1ITH|B  Chain B, Hemoglobin (Cyanomet) >gi|44305...  52.3  4e-08
```

Figure 7-15: The hit list in PSI-BLAST.

On the other hand, sometimes the road is a dead end. If, after the second iteration, every new hit is an alcohol dehydrogenase, that means it's time to give up! Unfortunately, no simple rule governs when to stop or when to continue. The only source of help is the annotation. Anything strange happening means one of two things: that you must stop — or that you have found something truly interesting.

REMEMBER

Bear in mind that you also have the obvious possibility (and even the obligation!) to uncheck clearly false matches between two iterations.

Choosing the right sequences

The main difficulty in PSI-BLAST is deciding which sequences you can keep from one iteration to the next. PSI-BLAST controls that decision automatically with the E-value threshold (Inclusion Threshold in the Format section) — but you can use the check box associated with each match to alter the automatic procedure manually.

This Inclusion Threshold is a mixed blessing, for the following reasons:

- If you set this threshold too low, PSI-BLAST doesn't catch any sequence.
- If you set this threshold too high, PSI-BLAST catches unrelated sequences.

and inspect the hits that are below the threshold. After each iteration, you can also add the hits that seem relevant (according to their annotations) and remove those that are selected but seem irrelevant. All this takes time — and it is a bit of an art — but sometimes it pays off.

Changing the query is also a means to check your results. If you use one of the matches as an initial query, you can reasonably expect that PSI-BLAST may find your original query after a few iterations.

Avoiding the confusion of multi-domain proteins

When you consider that a domain is something like a small autonomous protein that can fold on its own, it's no surprise that most proteins are multi-domains. The same domain may occur in very different proteins.

If your protein is multi-domain, PSI-BLAST can run into trouble because each domain competes with the others — and they may all recruit different sequences from unrelated proteins. The result you get from such a query is usually very difficult to interpret.

When this ambiguity happens, the best strategy is to cut your protein according to its domain structure — and then treat each sequence independently. You can find where the domains are by using the NCBI CD server or the EMBnet pfscan. (See Chapter 6 for more details.)

TIP

As a rule, if you didn't find any domain — and if your protein is larger than 200 amino acids — cut it into smaller pieces, each 200 amino acids long, and analyze them one by one. The snag in this strategy is that it leaves you with the boring task of putting together all the pieces of the puzzle. But sometimes that's where the fun begins.

Discovering and using protein domains with BLAST and PSI-BLAST

In Chapter 6, we show you how to look for known protein domains using the CD server (also known as *reverse-PSI-BLAST* or *rps-blast*). Even in that scenario, you can use BLAST to discover your own domains and use them to scan protein databases. In the BLAST dialect, a domain is named a *PSSM* (*P*osition *S*pecific *S*ubstitution *M*atrix).

Once your PSSM is ready, all you need do is a cut and paste it into the corresponding section of the advanced BLAST parameters form. (See Figure 7-11.)

When you do provide a PSSM, BLAST uses it in place of a single query sequence — and will compare it against every sequence in the database you choose.

Similarity Searches for Free over the Internet

The BLAST online documentation is terrific — you should use it as much as you can! The most informative pages are

- `www.ncbi.nlm.nih.gov/blast/producttable.shtml`

  This one's perfect for helping you choose the right flavor of BLAST and the right database.

- `www.ncbi.nlm.nih.gov/blast/blastcgihelp.shtml`

  This one tells you everything you need to know about each parameter.

If you've tried to run the examples in this chapter on the NCBI Web server, you know that sometimes there's just too much traffic to do useful work. Fortunately, BLAST servers keep popping up all around the world. Those we show in Table 7-6 constitute only a sample of the most robust and up-to-date servers.

Table 7-6 BLAST and PSI-BLAST Servers around the World

Country/ Continent	*Program*	*URL*
USA	BLAST/PSI-BLAST	`www.ncbi.nlm.nih.org/BLAST`
Europe	BLAST	`www.expasy.ch/tools/blast/`

(continued)

		bBLAST.html
Europe	BLAST	www.ebi.ac.uk/blast
Japan	BLAST/PSI-BLAST	www.ddbj.nig.ac.jp/search/blast-e.html

The NCBI-BLAST has a little sibling, known as WU-BLAST, where WU stands for Washington University. Specialists can spend hours arguing about why WU-BLAST is more sensitive and more gifted at inserting gaps than the NCBI-BLAST. The best argument here would probably be that you should try it for yourself (See Table 7-7) and join the debate . . .

Table 7-7 WU-BLAST

Address	*Description*
blast.wustl.edu/	The Home of WU-BLAST (no online server)
tigrblast.tigr.org/tgi/	WU-BLAST at TIGR, with a twist on bacterial genomes and nucleotide analysis
www.genome.wustl.edu/tools/blast/	blastn and tblastn, for digging into complete genomes
www.ebi.ac.uk/blast/	All flavors of BLAST and WU-BLAST
brassica.bbsrc.ac.uk/BrassicaDB/blast_form.html	Another complete WU-BLAST server

Although it's by far the most popular, BLAST isn't the only method available for searching databases. Three main alternatives to BLAST are

- **Smith and Waterman (SSEARCH):** It's slower — but arguably more accurate — than BLAST.
- **FASTA:** It's a bit slower than BLAST but allegedly more accurate when making DNA comparisons.
- **BLAT:** Use this for locating cDNA rapidly in a genome or finding close (mammalian-versus-mammalian) proteins in a genome.

None of these programs has any BLAST-type facility for filtering low-complexity regions.

Table 7-8 Alternative Methods for Homology Searches

Country/ Continent	*Program*	*Address*
USA	FASTA	`fasta.bioch.Virginia.edu/fasta`
Europe	FASTA	`www.ebi.ac.uk/fasta33`
Europe	SSEARCH	`www.ch.embnet.org/software/GMFDF_form.html`
Japan	SSEARCH/FASTA	`www.ddbj.nig.ac.jp/search/ssearch-e.html`
USA	BLAT	`genome.ucsc.edu`

Comparing Two Sequences

In This Chapter

- Making a dot plot over the Internet
- Identifying repeats in a protein sequence
- Finding out about local and global alignments
- Finding the best local alignments between your two sequences
- Getting a handle on the gap penalty and the substitution matrix

> *Your true value depends entirely on what you are compared with.*
>
> — Bob Wells

In this chapter, we show you some of the methods available for comparing two protein sequences — or two DNA sequences — with each other. In their own peculiar jargon, bioinformatics gurus call these analyses *pairwise comparisons* because they relate to a pair of sequences. They are extremely useful and can help you

- Convince yourself that two sequences are in fact homologous
- Find out that your sequences share a domain
- Identify the exact location of common features, such as disulfide bridges or catalytic active sites
- Compare a gene and its product

Typically, biologists use pairwise comparisons to refine database search results and carry out detailed analysis. It is common practice to use pairwise techniques after a database search in order to check whether some of the reported matches are really interesting or are just a flash in the pan.

In this chapter, we show you which method is best suited to your needs — and also show you how you can use the Internet to run a method on the sequences you're interested in.

The most delicate decision you make, when doing a pairwise analysis, is the choice of two sequences to compare. Choosing two sequences is a bit like arranging a boxing match between two opponents: The idea is to get the most exciting fight.

Don’t use pairwise comparison methods to discover a sequence that would be homologous to a sequence you already have. It takes too much time. If you want to compare your sequence with every other sequence in a database, use a database-search program such as BLAST. (See Chapter 7 for more on BLAST.)

If you’ve made your way through Chapter 7, you’re now in a position to argue that database-search programs merely do pairwise comparisons between a query sequence and all the sequences within a database — so there’s no real need to make extra pairwise comparisons, is there? But wait a minute. Although it’s true that programs like BLAST search databases through pairwise comparisons, these programs are optimized for speed, not for alignment accuracy. The programs we describe here are just the opposite: They are optimized for giving the most accurate possible result, which is why you want to apply them to carefully selected sequences.

Choosing the right sequences

A good reason to make a pairwise comparison between two sequences is a strong suspicion that these sequences are *homologous* — that is, they share a common ancestor. (See Chapter 7 for a complete description.) Such sequences often have similar 3-D structures and related functions.

The best way to find a sequence that’s homologous to a sequence you already have is to search a database with the BLAST program. Select your sequences according to the following (conservative) criteria:

- **DNA sequence:** At least 70 percent identity over more than 100 bases between the hit and the query, or an E-value lower than $10-^{4}$. (For more on E-values, see Chapter 7.)
- **Protein sequence:** More than 25 percent identity over more than 100 amino acids between the hit and the query, or an E-value lower than $10-^{4}$.

Unfortunately, these criteria are only an indication of a match, not a guarantee. If your hit has a score very close to the threshold, it may not be homologous to the query — or it may be so distantly related that aligning it correctly is difficult. This is where pairwise comparison methods come into the picture: They help you decide how meaningful a database hit really is.

Choosing a sequence from a database search output is all about making the right compromise between new information and certainty: You want a sequence that's similar enough to your query so you can be sure it's homologous. On the other hand, you want this sequence to be interesting from a biological point of view so the comparisons reveal something new to you.

In short, the object of your desire is an extensively annotated Swiss-Prot sequence that matches your query with an E-value much lower than 10-^{4}. In real life, we often have to compromise.

No written rule says you *must* select your sequences from a database search — doing so is simply very convenient. An alternative is to select your two sequences by using an experimental criterion: say, because they have the same function or other similar property. For this purpose, you may use the Sequence Retrieval System (SRS) that enables you to identify sequences by using keywords. (See Chapter 2 for more on this topic.)

If you have only one sequence, it is possible (and sometimes advisable) to make a pairwise comparison on this sequence alone. All you need to do is pretend that the second sequence is identical to the first one. This may seem a bit silly, but it's a good way to discover interesting features in your sequence, such as

- Repeated domains
- Regions with a small motif repeated many times (low complexity)
- Palindromes (portions of DNA that are repeated in different orientations) and potential secondary structures in RNA

Internal repeats can help elucidate the function of your protein. For instance, the discovery of a repeated region in the breast-cancer-susceptibility gene of type 1 (Brca1) made it possible to build a domain profile for this domain — and to identify the involvement of Brca1 in DNA repair mechanisms. Even by itself, this story makes a good case for not overlooking self-comparisons!

Choosing the right method

Comparing sequences is one of the most difficult tasks biologists ever came across. In fact, the problem is not entirely solved yet — which is why there are more solutions than one. Table 8-1 lists three methods and indicates the situations for which each is best suited.

	Finding long insertions/deletions Extracting portions of sequences to make a multiple alignment
Local alignments	Comparing sequences with partial homology: Making high-quality alignments Making residue-per-residue analysis
Global alignments	Comparing two sequences over their entire length: Identifying long insertions/deletions Checking the quality of your data Identifying every mutation in your sequences

It's a good strategy always to start with a *dot plot.* (We get into all the dot-plot details in the following section.) Dot plots are very powerful tools that give you an instant general picture. With a bit of experience, a simple glance at a dot plot immediately tells you what is really going on. It's an ideal tool for deciding on the next step of your analysis.

Yet, a dot plot is not enough for a fine-grained examination. Dot plots do not really produce alignments; they simply give generic indications. They don't tell unambiguously how amino acids or nucleotides correspond to one another between your two sequences.

To ferret out this important information, you have to make an alignment. There are two kinds of alignments: the global alignments (where the two sequences are aligned along their entire lengths) and the local alignment (where the program only aligns the most similar portions of the two sequences).

Global alignments are usually easier to interpret than local ones. However, it only makes sense to build a global alignment if your sequences are related over their entire lengths — and if they don't contain very long insertions or deletions. If you don't know how to choose between global and local alignments, always go for the local methods.

When you're comparing two sequences, start making a dot plot of each sequence against itself. This approach makes it easier to identify potential repeats within each sequence. With this information in hand, interpreting the dot plot of the two sequences is much simpler.

For instance, imagine that you want to compare these two sequences:

```
Sequence 1: THEFATCAT
Sequence 2: THEFASTCAT
```

Telling them apart isn't so obvious — they are almost identical, except for the extra *S* in Sequence 2. (Of course, that difficulty gets much worse with real sequences.)

To make a dot plot, take a piece of paper, draw a grid, and write out Sequence 1 along the top with one letter above each column. Next, write out Sequence 2 on the left side with one letter next to each line, as in the accompanying matrix.

This is where the game begins. Consider each cell one by one, and cross it if the top sequence and the side sequence contain the same

Of course, as in any tic-tac-toe game, the longest diagonals win! In this case, they indicate segments of similarity between your two sequences. The matrix below also makes it very easy to see that the two sequences are homologous over their entire length, with the exception of an extra residue in the vertical sequence.

```
..T H E F A T C A T
T X         X     X
H   X
E     X
F       X
A         X     X
S
T X          X    X
C              X
A         X      X
T X          X    X
```

The main difficulty when using the methods we list in Table 8-1 is the proper choice of parameters. Unfortunately, there is no absolute rule; producing the exact comparison you need is mostly a trial-and-error process. In the remainder of this chapter, we show you how to play with these parameters until they produce a result you can be satisfied with.

Making a Dot Plot

Dot plots are the simplest means of comparing two sequences. In fact, the dot plot — the kind you make using pencil and paper — is the only type of sequence comparison that you could do in the days when no one had true computer access. (See the sidebar "A DIY guide to dot plots.") Dot-plot techniques are closely related to sliding window methods — where a *window* is a portion of your sequence that you compare with other sequence portions of similar size. (For more on sliding windows, see Chapter 6). Dot plots have device available to you: your eyes. When it comes to identifying patterns such as diagonals, nothing beats the human eye, even when the signal is noisy. This is so true that any time you see a good signal on a dot plot, you can bet your life something interesting is going on — even if the statistics are too “smart” to see anything!

Choosing the right dot-plot flavor

In this section, we introduce a very powerful tool named Dotlet. Dotlet is very convenient because it’s easy to use, free of charge, doesn’t need any installation, and runs on (almost) any computer that has access to the Internet.

Dotlet is ideal for sequences with lengths of less than 10,000 amino acids or nucleotides. Thus it can be helpful for most proteins but is restricted to small DNA sequences. If you want to look at longer sequences online, you will have to use Dnadot (Table 8-2), a tool designed for fast dot-plotting between long sequences.

If your sequences are longer than 100,000 characters, no adequate tool is available online — you need to install a more powerful version of Dotlet on your computer. At least two such packages are available for free: Dotter and Dottup. You can find their respective specifications (and URLs) in Table 8-2.

Table 8-2 Various Flavors of Dot-Plot Programs

Name	*Used For*	*Range*	*URL*	*Platforms*
Dotlet	Proteins, DNA	10,000	`www.ch.embnet.org`	All (Java)
Dnadot	Proteins, DNA	100,000	`arbl.cvmbs.colostate.edu/molkit/dnadot/`	All (Java)
Dotter	Proteins, DNA	100,000	`www.cgr.ki.se/cgr/groups/sonnhammer/Dotter.html`	Unix, Linux, Windows
Dottup	Complete genomes, DNA	>100,000	`www.emboss.org`	Unix, Linux

and then runs on your own machine. The nice thing about it is that you don't need to install anything, as long as you have Java running (which most Internet browsers do by default these days).

Dotlet isn't a complicated program, but it contains many features. If you don't use these features properly, you may pass very close to the result you need but miss it. The general rule when using Dotlet is that you want to find a clear signal — and that you should keep playing with the parameters until you get one.

In the upcoming steps list, we show you some examples of what we mean by the term *clear signal* and give some tips on how best to obtain one.

Downloading Dotlet

You can download Dotlet with one click of a mouse, simply by pointing your browser to `www.isrec.isb-sib.ch/java/dotlet/Dotlet.html`. When you open this page, your browser also downloads the Dotlet program. This may take a few minutes if you're on a slow connection.

When the download is finished, your browser displays a page like the one shown in Figure 8-1.

If your browser does not display this page properly, it means that it can't run Java; you may need to update to the latest version of Internet Explorer or Netscape in order to get Java running on your computer.

Dotlet is not a server but a Java applet. Everything Dotlet does, it does on your own computer — not on the EMBnet server. For you, this has three consequences:

- You can use Dotlet offline after you download it once. This is useful if you work on a laptop that isn't always connected.
- The speed of the program depends on your own computer. The faster your computer, the faster Dotlet runs. Differences become apparent with sequences longer than 1,000 residues.
- When you use Dotlet, *you are not sending your sequences anywhere!* Each sequence you compare with Dotlet stays on your computer and isn't sent to the EMBnet server. If you work in a company that's at all interested in protecting proprietary information, you know this is a VERY GOOD thing!

  To make sure Dotlet isn't communicating with any server when you use it, choose File➪Work Offline from your browser menu after you've downloaded Dotlet.

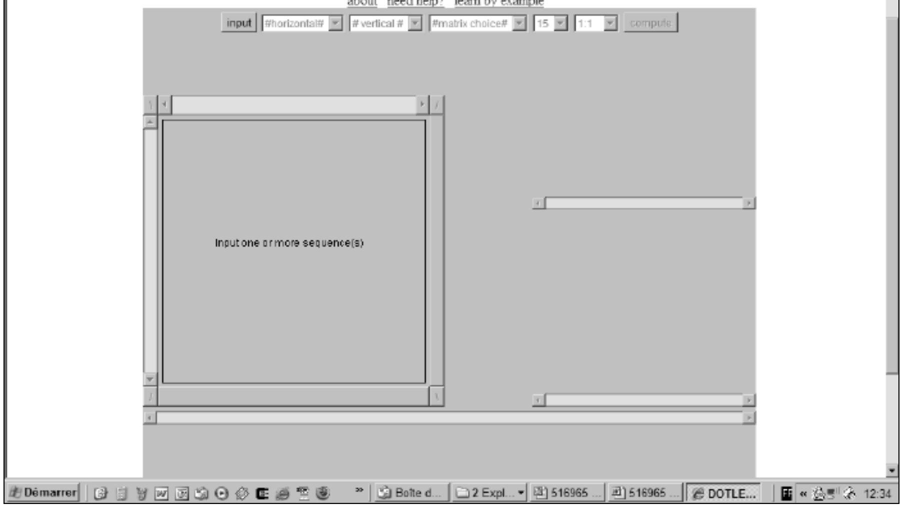


Figure 8-1: The Dotlet page.

Entering your sequence in Dotlet

In the following steps list, we compare two sequences that contain a kinase domain. We're going to assume that you've already got Dotlet up and running in your browser (if not, the address is `www.isrec.isb-sib.ch/java/dotlet/Dotlet.html`):

1. **Open a new browser window by pressing Ctrl+N on your keyboard.**

   You need access to two browser windows, which is why we start with this step.

2. **Type the address** `www.expasy.ch/cgi-bin/get-sprot-fasta?P05049` **into the new window you opened in Step 1 and then press Enter or Return.**

   Type in the exact URL; do not omit the `?`. This URL gives you a direct access to Swiss-Prot if you know the ID number of your sequence. For our example, the ID number is P05049.

   After a bit of a wait, a new window appears, displaying the sequence associated with the ID number you just entered.

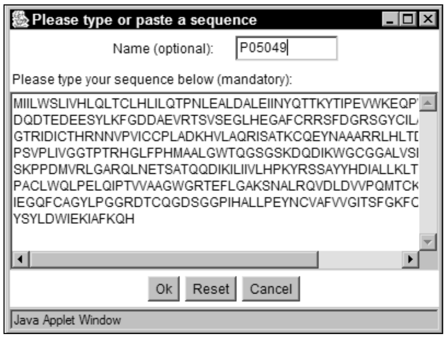
5. **Use Ctrl+V to paste your sequence into the pop-up window and then enter a title.**

   You must enter your sequence in raw format (only the sequence) as shown in Figure 8-2.

   The Name box is where you can enter the title of your sequence on the dot plot if you so desire.

6. **Press the OK button in the Sequence Window.**

   When you press the OK button, the window disappears; this means your sequence is now loaded in Dotlet.



Please type or paste a sequence

Name (optional): P05049

Please type your sequence below (mandatory):

```
MIILWSLIVHLQLTCLHLILQTPNLEALDALEIINYQTTKYTIPEVWKEQP
DQDTEDEESYLKFGDDAEVRTSVSEGLHEGAFCRRSFDGRSGYCIL
GTRIDICTHRNNVPVICCPLADKHVLAQRISATKCQEYNAAARRLHLTD
PSVPLIVGGTPTRHGLFPHMAALGWTQGSGSKDQDIKWGCGGALVSI
SKPPDMVRLGARQLNETSATQQDIKILIIVLHPKYRSSAYYHDIALLKLT
PACLWQLPELQIPTVVAAGWGRTEFLGAKSNALRQVDLDVVPQMTCK
IEGQFCAGYLPGGRDTCQGDSGGPIHALLPEYNCVAFVVGITSFGKFC
YSYLDWIEKIAFKQH
```

Figure 8-2: The sequence window in Dotlet.

If you want to compare a sequence only with itself, go directly to Step 13.

7. **Type the address** `www.expasy.ch/cgi-bin/get-sprot-fasta?P08246` **into the new window you opened in Step 1, and press Enter or Return.**

   This is the second sequence you want to load into Dotlet.

8. **Copy the sequence displayed in your browser.**

9. **In the Dotlet window (refer to Figure 8-2), click the Input button.**

10. **Paste your sequence using Ctrl-V in the pop-up window that appears and enter a title.**

the two sequences you want to compare from the two drop-down windows nearest to the Input button, as shown in Figure 8-3.

Figure 8-3 shows you how to use the pull-down menus to choose the sequences you want to compare. The names in the menus are those you provided in the Title box in Steps 5 and 10.

- The sequence you choose in the first pull-down menu is the horizontal sequence (top of the dot plot, left to right).
- The sequence you choose in the second menu is the vertical sequence (left side of the dot plot, top to bottom).

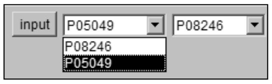


Figure 8-3: The sequence pull-down menus in Dotlet.

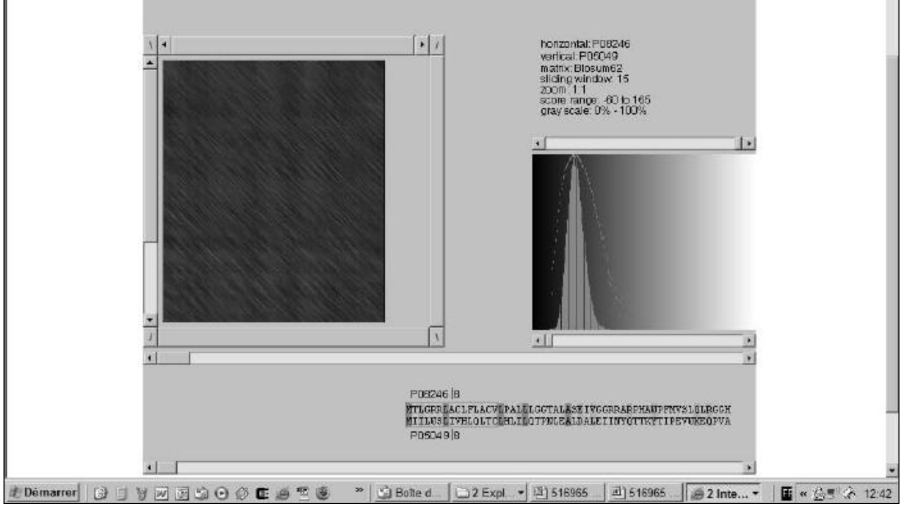
13. **In the Dotlet window, click the Compute button at the far right.**

    Dotlet never takes action automatically (except when you change the threshold). Any time you change one of its parameters, you must inform Dotlet by clicking the Compute button.

    The output looks like what's shown in Figure 8-4. This figure shows Dotlet's three windows:

    - **The dot-plot window:** The large window on the left contains your dot plot.
    - **The threshold window:** This small window on the right is the one you must use to tune the sensitivity of your dot plot.
    - **The alignment window:** This is the long narrow window below that shows which pair of residues from the two sequences corresponds to each dot in the dot-plot window.

The default output of Dotlet is usually difficult to interpret. To make it talk some sense, you have to refine it — which we show you how to do in the next section.



horizontal: P08246
vertical: P05049
matrix: Blosum62
sliding window: 15
zoom: 1:1
score range: -60 to 165
gray scale: 0% - 100%

P08246 |8
MTLGRRLACLFLACVLPALLLGGTALASEIVGGRRARPHAWPFMVSLQLRGGH
MIILUSLIVHLQLTCLHLILQTPNLEALDALEIINYQTTKFTIPEVUKEQPVA
P05049 |8

Figure 8-4: Default output of Dotlet.

Fine-tuning your Dotlet

The sequence of instructions we show here isn't necessarily optimal, but it follows a top-down approach. We start with settings that you can roughly adjust, and finish with settings that you must fine-tune to yield an informative dot plot.

1. **Set the zoom factor.**

   By default, Dotlet has a zoom factor of 1:1 — which means that 1 pixel corresponds to 1 residue.

   If your sequences are long, you may not see the entire dot plot. This is exactly what happens in Figure 8-4. You can tell this is so because the scroll bar down the left side is *active* — you'd need to use the rectangular box in the scroll bar to scroll down to see the rest of the display. To see the entire dot plot, do the following:

   a. **In the Zoom selector — the pull-down menu to the left of the Compute button — choose 1:2, as shown in Figure 8-5.**

   b. **Click the Compute button.**

      Dotlet now displays a complete dot plot of your two sequences.

   Seeing the overall picture is useful, even if it looks a bit small. After you get a feeling for the regions that are interesting in your dot plot, you can always zoom back.

ple, you have to change the default value as follows:

   a. **Choose 51 from the Window pull-down menu (the menu to the left of the Zoom selector).**

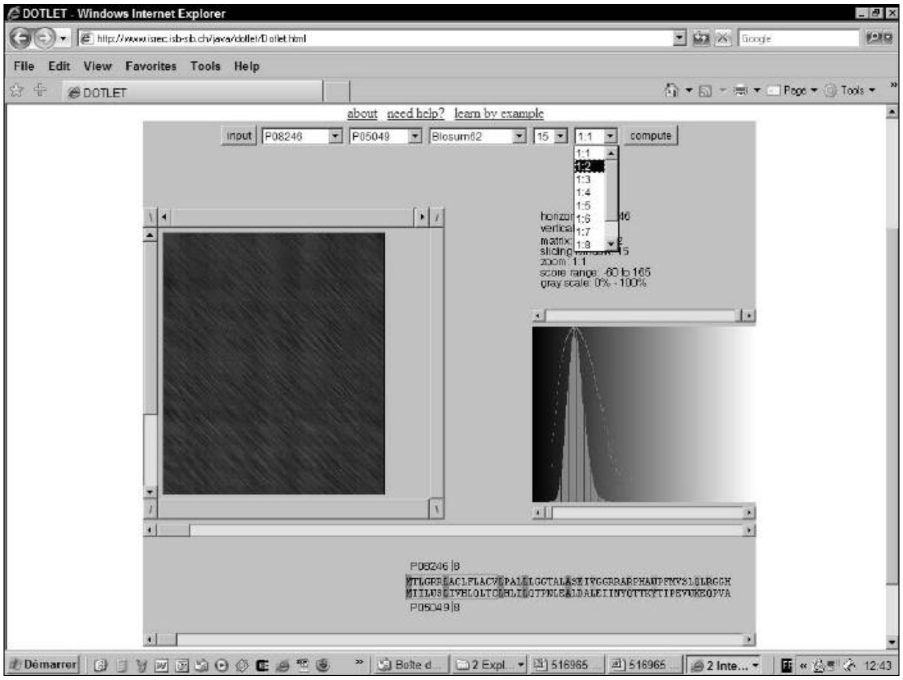
   b. **Click the Compute button.**

WARNING!

After changing the window size, you MUST also tune the threshold (see Step 3). There is little point in changing the window size if you don't adjust the threshold accordingly.

3. **Adjust the threshold.**

   The *threshold* is a value that defines the color of each dot in the dot-plot window. Dots with a score above the threshold are white; those with a score below the threshold are black. The threshold is by far the most powerful and the most delicate parameter when you're making a dot plot. Fortunately, Dotlet lets you adjust the threshold dynamically and see the effect in real time. (You don't even have to click the Compute button!)


DOTLET - Windows Internet Explorer
http://www.isrec.isb-sib.ch/java/dotlet/Dotlet.html
Google
File Edit View Favorites Tools Help
DOTLET
Page
Tools
about need help? learn by example
input
P08246
P05049
Blosum62
15
1:1
compute
1:1
1:2
1:3
1:4
1:5
1:6
1:7
1:8
Démarrer
Boîte d...
2 Expl...
516965
516965
2 Inte...
12:43

```
horizon...46
vertical...
matrix:...2
slicing window: 15
zoom: 1:1
score range: -60 to 165
gray scale: 0% - 100%
```

```
P08246 |8
MTLGRRLACLFLACVLPALLLGGTALASEIVGGRRARPHAWPFMVSLQLRGGH
MIILVSLIVHLQLTCLHLILLGTPNLEALDALEIINYQTTKYTIPEVWKEQPVA
P05049|8
```

Figure 8-5: Choosing the right zoom factor in Dotlet.

- Long windows make clean dot plots. They make sure that two similar amino acids (or nucleotides) only yield a dot if the amino acids (or nucleotides) around them are also similar. Gurus say that long windows make the dot plot more *stringent.* Of course, being too stringent is just as bad as not being stringent enough. In order to achieve the right equilibrium, you can use the following simple guidelines:
- The size of a window should be within the same range as the size of the elements

teins, a size of 50 amino acids or higher is appropriate.

- Shorter windows are more sensitive but bring some noise with them. They may help if you're looking at protein domains that are very distantly related.
- It's convenient to start with a large window and narrow it a little until the signal you're looking for appears.

On the top of the threshold window — the window on the right side for the Dotlet display in Figure 8-4 — there is a little horizontal scroll bar with a cursor. This is the *top cursor.* On the bottom of this same window, there is another small horizontal scroll bar with another small cursor. This is the *bottom cursor.*

The horizontal axis in the threshold window represents the score. Low scores are to the left, high scores to the right. The big peak on the curve shows you the most common score for two residues chosen randomly.

You can imagine the threshold as two vertical lines going through the top and the bottom cursor. Its position controls the appearance of the dot-plot window (the large window on the left in Figure 8-4). In this dot-plot window

- Dots with a score below the bottom cursor threshold are black.
- Dots with a score above the top cursor threshold are white.
- Dots with a score between the two cursors' thresholds are gray.

The game is all about setting the threshold so interesting residues appear as white dots on a black background in the dot-plot window on the left. You can use the following procedure to achieve this effect:

a. **Drag the bottom cursor in the threshold window entirely to the right.**

   This makes the dot plot on the left entirely black.

When the top cursor and the bottom cursor are on the top of each other, dots in the dot-plot window are either black or white. If you shift one of the cursors to the left or to the right, the dots whose scores are between these two thresholds appear in gray. This can help smooth the plot and make it look nicer for publication.

If you want to see black dots on a white background, shift the top cursor slightly to the right of the bottom cursor. This inverts the color chart.

4. **Save your dot plot.**

   Saving your results is the most delicate part of using Dotlet. Not all browsers can print these results — and we didn't find one that could save the display as a file. As far as we know, there are only two ways to keep a record of your display:

   Print the browser content into a PDF file. To do so, you must have a PDF printer installed on your computer.

   Use the ol' screen-capture technique. You can use it with any graphic display that gives you trouble when trying to save. Follow this procedure to do a screen capture.

   a. **Press the PrntScrn button on your keyboard (usually top right).**

   b. **Start a Power Point presentation.**

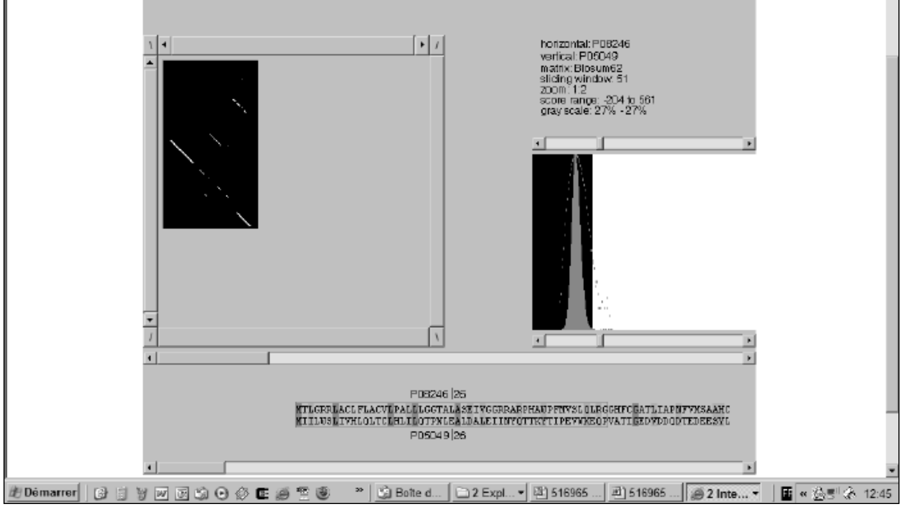
   c. **Press Ctrl+V to cut and paste your screen-capture into the presentation.**

      You can now print your dot plot or include it in any type of document.

Interpreting your results

P05049 and P08246, the two proteins we chose for the previous steps list (in the section "Entering your sequence in Dotlet"), are very distantly related. If you use BLAST to find one with the other, you get a match with an E-value of 10-^{4}. If you have been through Chapter 7 — which introduces you to the intricacies of BLAST — you already know that getting really excited over such a match isn't a good idea. It could even be a false positive.

Knowing that P05049 is a serine protease, it would be interesting to check the protease activity of P08246 in the wet lab. The problem is that when you launch a costly experiment on the basis of such a weak BLAST match, you're definitely taking a chance. If you fail, the probabilities are high that your boss will ask the lethal question, "What in the world where you thinking?!"



horizontal: P08246
vertical: P05049
matrix: Blosum62
sliding window: 51
zoom: 1:2
score range: -204 to 561
gray scale: 27% -27%

```
P08246 |25
MTLGRRLACLFLACVLPALLLGGTALASEIVGGRRARPHAWPFMVSLQLRGGHFCGATLIAPNFVMSAAHC
MIILVSLIVHLQLTCLHLILQTPNLEALDALEIINYQTTKYTIPEVWKEQPVATIGEDVDDQDTEDEESYL
P05049 |26
```

Figure 8-6: Dotlet with a proper parameter setting.

This is the reason why — before doing this experiment — it would be reassuring to make sure that P05049 and P08246 are genuinely homologous in the serine protease region.

Making a dot plot is a good way to answer this question. A dot plot like the one we produced in the previous steps list is very much to the point. Figure 8-6 makes a good case for the existence of a conserved domain between the two proteins. You can make sure of this by looking at their Swiss-Prot annotation (accessible at `www.expasy.ch`). Swiss-Prot says that P05049 and P08246 both contain a serine protease domain in the region that the dot plot indicates.

Of course, in real life you may not have a Swiss-Prot annotation for the two proteins. In fact, in the real world, you may not have any annotation at all! That should not deter you from making a dot plot. Even if you don't know its function, the identification of a conserved domain between two protein sequences is a good way to locate the specific region responsible for a function.

Doing biological analysis with a dot plot

The preceding example shows you how to spot a distantly related domain shared by two proteins. Dot plots give very characteristic representations of

Internet.”

Identifying tandem repeats

Proteins commonly contain a small domain repeated many times over. It seems that internal duplication is a tool often used by evolution to create new proteins or make them function more efficiently. Domains that you may often find duplicated include the Fibronectin domain (we like to refer to it as “molecular Velcro”), EF-hands that bind calcium, or Zn fingers involved in DNA binding.

Dot plots constitute by far the best way to spot repeated domains. In Figure 8-7, we show you the YE73 human protein sequence. YE73 is a potential human transcription factor (involved in RNA transcription) that contains 13 very conserved Zn finger domains in tandem.

To analyze this sequence, follow these steps:

1. **Point your browser to** `www.expasy.ch/cgi-bin/get-sprot-fasta?Q9P255`**.**

   Q9P255 is the ID number for the YE73 human protein sequence.

2. **Enter the sequence into Dotlet, following the procedure we outline in the section “Using Dotlet over the Internet.”**

3. **Set the threshold and window settings so they match those in Figure 8-7.**

   You want a window size of 21, a zoom of 1:2, and a gray scale for the top and bottom cursors of the threshold window set to 39%.

   The pattern in Figure 8-7 is typical of tandem repeats. Notice the following features:

   - The main diagonal represents the sequence against itself.
   - Repeats appear as long continuous diagonals above and below the main diagonal.
   - The diagonals are evenly spaced.
   - The collection of diagonals is bound by the shape of a square.

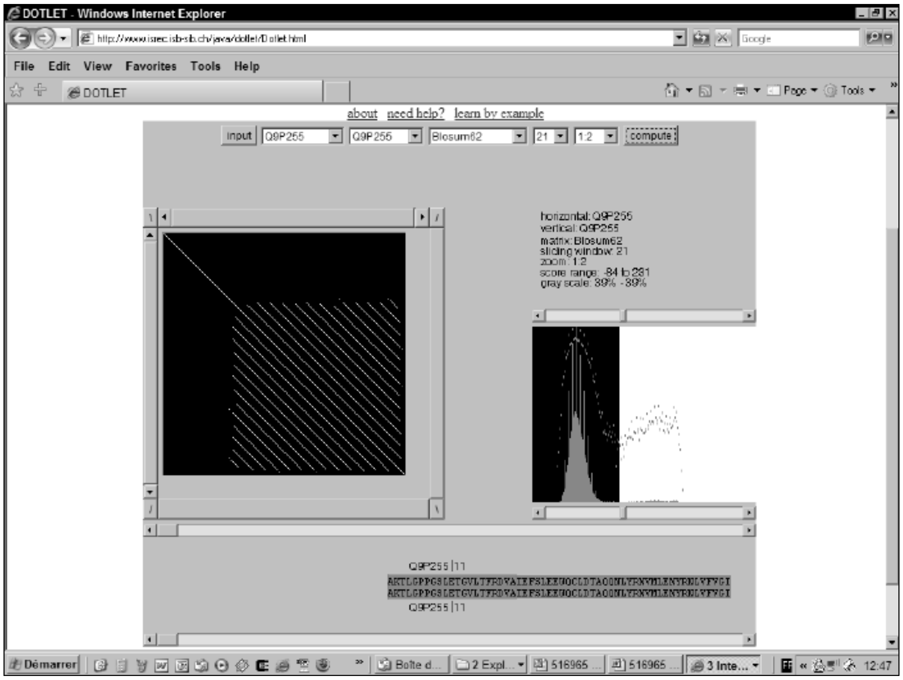
Anytime you see this type of graphic pattern, you know you’re looking at tandem repeats. If this is the case, you can use this collection of diagonals to make three interesting deductions:

YE73 is an extreme example because it contains highly conserved repeats. Life isn't always so simple — and, in most cases, the repeated domains you come across aren't so similar (because of accumulated mutations). Proteins that contain distantly related repeats produce more complicated dot-plot patterns. In these proteins, each repeat unit doesn't necessarily recognize every other unit.

Figure 8-8 shows you such an example. Tf3a is a transcription factor that also contains Zn finger domains in tandem. You can find the sequence at

```
www.expasy.ch/cgi-bin/get-sprot-fasta?P03001
```

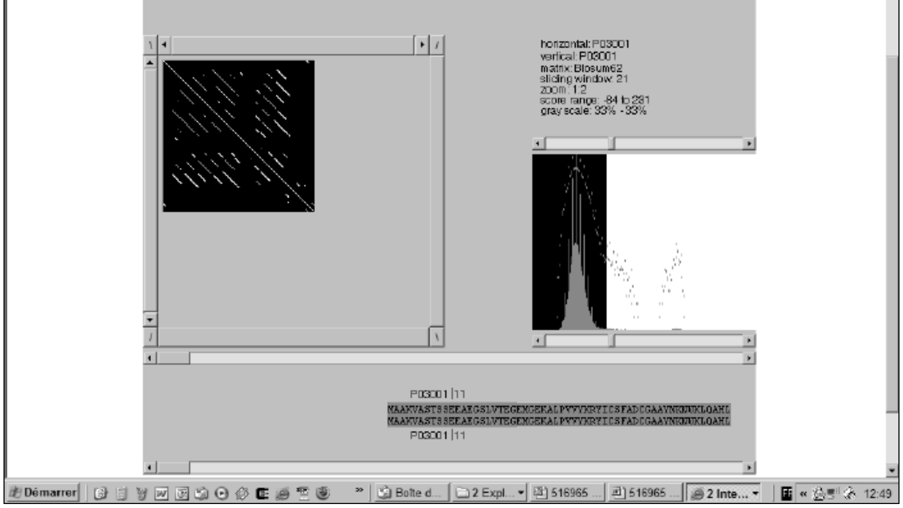
Tf3a tells the story of Zn finger domains that have been duplicated and have diverged since then. These Zn fingers are less conserved, and show a pattern much less regular than what we obtained with YE73.



horizontal: Q9P255
vertical: Q9P255
matrix: Blosum62
sliding window: 21
zoom : 1:2
score range: -84 to 231
gray scale: 39% - 39%

```
Q9P255 |11
ARTLGPPGSLETGVLTFRDVAIEFSLEEWQCLDTAQQNLYRNVMLENYRNLVFVGI
ARTLGPPGSLETGVLTFRDVAIEFSLEEWQCLDTAQQNLYRNVMLENYRNLVFVGI
Q9P255 |11
```

Figure 8-7: Looking at tandem repeats with Dotlet.



```
horizontal: P03001
vertical: P03001
matrix: Blosum62
sliding window: 21
zoom: 1:2
score range: -84 to 231
gray scale: 33% - 33%
```

```
P03001 |11
MAAKVASTSSEEAEGSLVTEGEMGEKALPVVYKRYICSFADCGAAYNKNWKLQAHL
MAAKVASTSSEEAEGSLVTEGEMGEKALPVVYKRYICSFADCGAAYNKNWKLQAHL
P03001 |11
```

Figure 8-8: Looking at the poorly conserved tandem repeats of Tf3a with Dotlet.

Tandem domains are not the only form of repeated domains in proteins or DNA. It is safe to assume that any kind of repetition you can think of probably occurs in proteins or in nucleic acids. If it has not yet been found, you can bet it's about to be discovered!

The signature of a repetition is always the same: It's a diagonal that lies off the main diagonal when you compare the sequence with itself.

Repeated domains can help you elucidate the function of your protein. If you find a repeated domain with no known function — and no similarity with any characterized protein — you can do the following:

1. **Extract each repeat unit.**
2. **Make a multiple alignment of the domains.**

   See Chapter 9 for more about how to do this.
3. **Identify conserved positions in the domain.**
4. **Turn your domain into a PROSITE pattern or a profile.**

   See Chapter 6 for more on PROSITE patterns.
5. **Scan Swiss-Prot in order to check whether this pattern is associated with a function.**

Low-complexity regions often have biological functions like protein-protein interaction (Leucine Zipper) or nonspecific DNA/RNA binding (ARG-rich domains).

The good news about low-complexity regions is that seeing them on a dot plot is really easy. When you compare a sequence with itself, low-complexity regions pop up as squares. You can see one of these in Figure 8-9.

To reproduce the results you see on Figure 8-9, follow these steps:

1. **Point your browser to** `www.expasy.ch/cgi-bin/get-sprot-fasta?P21997`**.**

   P21997 is the ID number for the sulfated surface glycoprotein 185.

2. **Enter the sequence into Dotlet, following the procedure we outline in the section "Using Dotlet over the Internet."**

3. **Set the settings so they match those in Figure 8-9.**

   You need a window setting of 7, a zoom setting of 1:2, and a gray scale for the top and bottom cursors of the threshold window set to 44%.

Analyzing nucleic acids with Dotlet

By nature, dot plots are well suited for mapping genes. Unfortunately, Dotlet can be a bit slow for such applications, which routinely require the comparison of sequences longer than 10,000 residues.

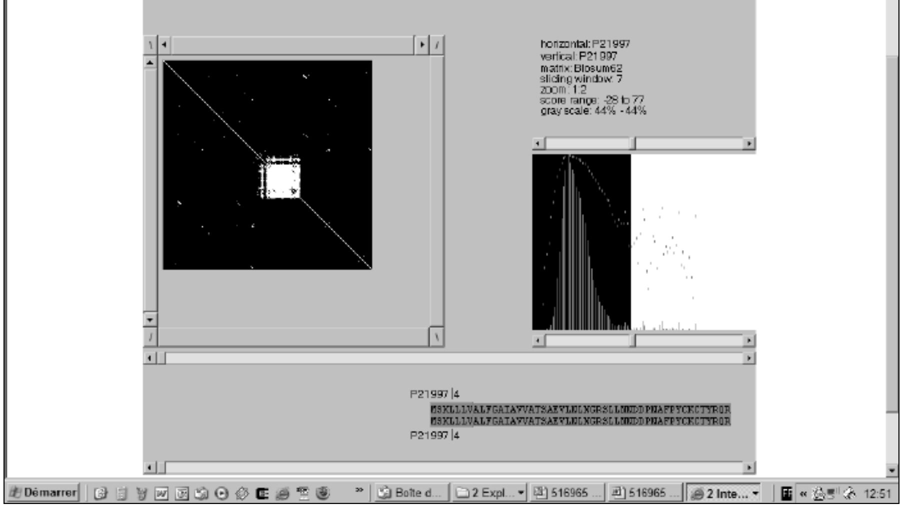
TIP

Dotlet contains two useful features for analyzing nucleotide sequences:

- **If you compare a protein sequence and a nucleotide sequence:** Dotlet automatically translates the nucleotide sequences into its three possible protein-reading frames. This way you can pinpoint exon/intron boundaries.
- **If you compare a nucleotide sequence with itself:** Dotlet automatically replaces one of the sequences with its complementary sequence. This way, if your sequence contains two complementary strands (like a hairpin stem in an RNA structure), they appear as diagonals going from the top-right corner toward the bottom-left (perpendicular to the main diagonal).

If you want to use Dotlet for more refined sequence analysis, we recommend the excellent online tutorial produced by Dotlet authors, available at

```
www.isrec.isb-sib.ch/java/dotlet/dotlet_examples.html
```



horizontal: P21997
vertical: P21997
matrix: Blosum62
sliding window: 7
zoom: 1:2
score range: -28 to 77
gray scale: 44% - 44%

```
P21997 |4
    MSKLLLVALFGAIAVVATSAEVLELNGRSLLMDDPNAFPYCKCTYRQR
    MSKLLLVALFGAIAVVATSAEVLELNGRSLLMDDPNAFPYCKCTYRQR
P21997 |4
```

Figure 8-9: Identifying low-complexity segments with Dotlet.

Making Local Alignments over the Internet

If you've gone through the dot-plot section of this chapter, you know that making a dot plot is an ideal way to get an overview of the relationship between your sequences. Nonetheless, dot plots aren't predictive methods: They show a signal but don't tell you what it means. To make this interpretation, you need a method that produces an alignment.

REMEMBER

There are two kinds of alignments: *global* (where the two sequences are aligned over their entire lengths) and *local* (where the program only aligns the most similar portions of your two sequences and ignores the rest).

TIP

If you have a dot plot showing that your two sequences are related over their entire lengths, you may not need to make a local alignment, and you can go directly to the global alignment section of this chapter. If you're not sure, stay here!

- To analyze repeated elements within a single sequence

Local alignment methods do exactly what their name implies: You give them two sequences, and they output an alignment of the two most similar portions of these sequences.

The nice thing about local alignment methods is that they automatically get rid of the amino acids or nucleotides that they can't align.

In theory, a local alignment should correspond to one of the diagonals that appear on the dot plot. In practice, however, things aren't so simple; you may see a signal on the dot plot that doesn't correspond to any local alignment, and sometimes local alignment programs report alignments that don't appear on the dot plot. Only when the dot plot and the alignment agree perfectly can you be sure.

Another rule with sequence alignments is that you should trust them only when they are *clearly correct.* (In this section, we show you what we mean by "clearly correct.") The only situation where you can use a suspicious alignment is when structural or experimental information exists that supports this alignment.

Choosing the right local-alignment flavor

Two types of methods exist for making local pairwise alignments: a fast, heuristic method named BLAST and a slower, more accurate one named Lalign. Table 8-3 lists the pros and cons of these two methods so you can select the one that suits you best.

The version of BLAST that compares two sequences is the same one you can use for searching a database. It's adapted so you can restrict it to two sequences only, but it doesn't generate alignments different from those BLAST reports when it's searching complete databases. In short, if you want to check a hit, BLASTing two sequences can't tell you anything more than the database search. You can access this special flavor of BLAST at

```
www.ncbi.nlm.nih.gov/blast/bl2seq/wblast2.cgi
```

Size of sequences you can use	Very long sequences	Shorter sequences
Score	E-value	Matrix score and E-value
Alignments	Reports the single best	Reports the ten best (or more)
Sequence type	Best with DNA	Best with proteins

Using Lalign to find the ten best local alignments

The main problem when using BLAST is that this program only returns one alignment: the best-scoring one. This is very useful for searching databases, but it may not be so interesting if the region of the sequences you're interested in happens to belong to a *good* local alignment but not to the *best* local alignment. It's a bit like a music store that would only sell you one CD — the number-one bestseller of the week.

Lalign is the ideal complement to BLAST. It is slower but more accurate, and it returns as many local alignments as you want (the best scoring one, the second best scoring, and so on, up to a number that you specify). Don't worry about the speed issue. On sequences shorter than a thousand symbols, you may not notice any difference between the bl2seq server and Lalign.

In general, Lalign is great when it comes to analyzing complicated proteins full of repeats. In the following steps list, we use Lalign to extract local alignments from two distantly related sequences that both contain a serine protease domain:

1. **Point your browser to** `www.ch.embnet.org/software/LALIGN_form.html`**.**

   The Lalign page of `ch.EMBnet.org` appears.

2. **Choose Local from the alignment method options, as shown in Figure 8-10.**

   You can use the Lalign interface to do local and global alignments.

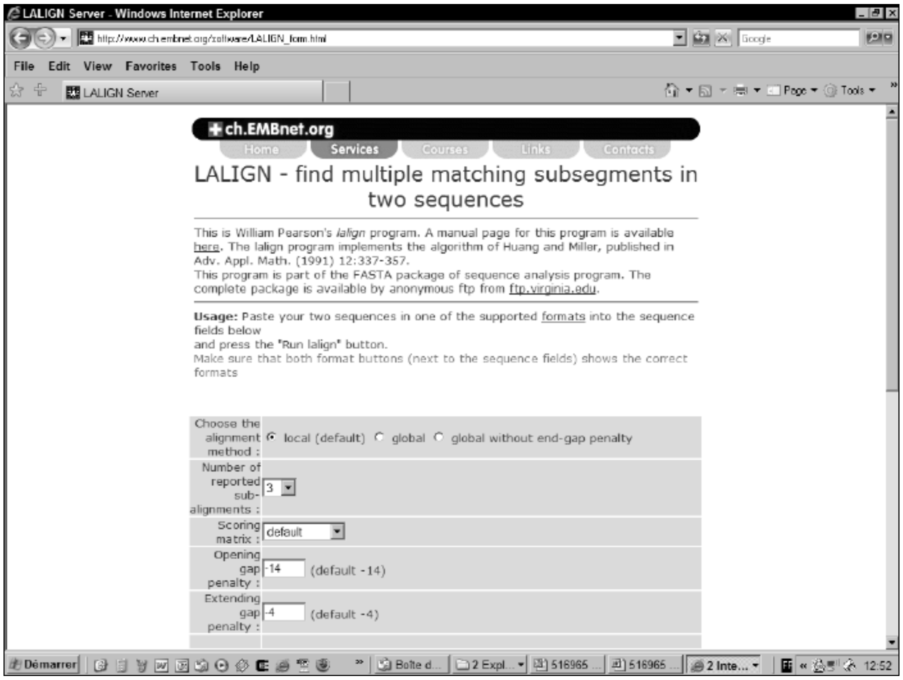
The *substitution matrix* controls the cost of the mutations when Lalign aligns the sequences.

The default matrix is an identity matrix for DNA and a BLOSUM45 for proteins. In most cases, this default is entirely appropriate. If you want to change the matrix, bear in mind that high BLOSUM indexes make Lalign more stringent — and result in shorter alignments. Low PAM indexes have the opposite effect.

5. **Set the gap-opening penalty.**

   The *gap-opening penalty* defines the cost for opening a gap in one of the sequences.

   If you set the gap-opening penalty higher than its default value, local alignments that contain gaps may be split into several shorter alignments. There is no simple rule to predict the optimal value for the gap penalty.


LALIGN Server - Windows Internet Explorer
http://www.ch.embnet.org/software/LALIGN_form.html
File Edit View Favorites Tools Help
LALIGN Server
ch.EMBnet.org
Home
Services
Courses
Links
Contacts
Démarrer
Boîte d...
2 Expl...
516965 ...
516965 ...
2 Inte...
12:52

LALIGN - find multiple matching subsegments in two sequences

This is William Pearson's *lalign* program. A manual page for this program is available here. The lalign program implements the algorithm of Huang and Miller, published in Adv. Appl. Math. (1991) 12:337-357.
This program is part of the FASTA package of sequence analysis program. The complete package is available by anonymous ftp from ftp.virginia.edu.

Usage: Paste your two sequences in one of the supported formats into the sequence fields below
and press the "Run lalign" button.
Make sure that both format buttons (next to the sequence fields) shows the correct formats

Choose the alignment method :	● local (default) ○ global ○ global without end-gap penalty
Number of reported sub-alignments :	3
Scoring matrix :	default
Opening gap penalty :	-14 (default -14)
Extending gap penalty :	-4 (default -4)

Figure 8-10: The Lalign home page.

The *gap-extension penalty* is an extra penalty proportional to the length of the gap. This penalty is added to the gap-opening penalty to yield the complete gap cost. It must be about ten times lower than the gap-opening penalty.

When you compare distantly related sequences, a high gap-opening penalty and a very low gap-extension penalty often yield the best results. They indicate that gaps should be penalized more on the basis of their existence than of their length.

7. **Choose Swiss-Prot ID or AC from the Input Sequence Format pull-down menu.**

   You'll need to scroll down the Lalign page to get to the menu.

8. **Enter** P05049 **in the first sequence box.**

   P05049 is the accession number of the snake serine protease. If you want to use a sequence that has no accession number, paste it in the window — and make sure that you chose the correct format in Step 7.

9. **Set the second format selector to Swiss-Prot ID or AC.**

10. **Enter** P08246 **in the second sequence box.**

    P08246 is the accession number of the human leucocyte elastase.

11. **Click the Run Lalign button.**

    The computation should be relatively fast (less than one minute). If the program doesn't return any results, check to make sure you've submitted the right parameters.

    Lalign returns the results in the form of an HTML document.

12. **Save your results.**

    Use the File➪Save As option of your Web browser to keep this result in a file.

Interpreting the Lalign output

Lalign reports the number of local alignments you have specified — as in the preceding Step 3 — sorted according to their score. An interesting property of Lalign is that it only reports *nonoverlapping* alignments.

This means that, in the Lalign output, you can find two amino acids or nucleotides aligned together only once. They can appear in several alignments alignment are the following features:

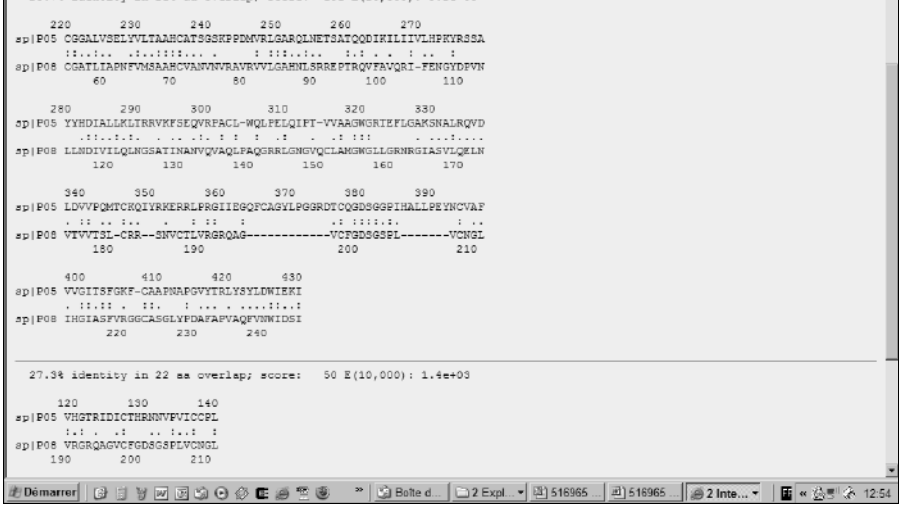
- **The percent identity:** The proportion of identical residues aligned with one another. For instance, you can see in Figure 8-11 that the best local alignment has a percent-identity score of 25.7 percent; the second-best score is 27.3 percent.
- **The local alignment length (Overlap):** This is the total length of the local alignment.
- **The score:** This score sums up the cost of the gaps and substitutions, as given by the substitution matrix and the gap penalties. Generally speaking, the higher the score, the better the alignment — yet be aware that the absolute value has no clear meaning. The E-value is a better indicator of the alignment's quality.
- **The E-value:** This value tells you how many times you could have expected to find such a good alignment by chance, given your two sequences. Be aware that this E-value is much less meaningful than the one BLAST reports when searching a database. A good E-value must be below 10-^{4}.

The alignment itself contains three types of information:

- **The residue index on the line above the sequence.** The residue corresponding to an index is the one below the last digit of this index.
- **The alignment itself, with gaps represented by dashes.**
- **Identity and similarity.** In the line between the two aligned sequences, the (_) symbol means identity, while the (.) symbols mean similarity. Two residues are similar when their substitution score is greater than 0.

The first alignment reported corresponds to the conserved serine protease domain. On its own, this alignment isn't really convincing: It contains less than 26 percent identity over about 200 residues.

To be really convinced, we would need a much higher similarity (at least close to 30 percent) and a better (lower) E-value. The reason we may trust this alignment is because it is consistent with the signal we previously saw on the dot plot. (Refer to Figure 8-7.) The fact that these two different analyses give compatible results (Dotlet and Lalign) makes a good case for the existence of a conserved serine protease domain in our proteins.



```
     220       230       240       250       260       270
sp|P05 CGGALVSELYVLTAAHCATSGSKPPDMVRLGARQLNETSATQQDIKILIIVLHPKYRSSA
       ::..:..  .:..::::... .      : :::..:..   :.: . .  : ..  :
sp|P08 CGATLIAPNFVMSAAHCVANVNVRAVRVVLGAHNLSRREPTRQVFAVQRI-FENGYDPVN
           60        70        80        90       100       110

     280       290       300       310       320       330
sp|P05 YYHDIALLKLTRRVKFSEQVRPACL-WQLPELQIPT-VVAAGWGRIEFLGAKSNALRQVD
         .::..:.:.  . .. .:. : :  :  .:   .  .: :::      . ...:....
sp|P08 LLNDIVILQLNGSATINANVQVAQLPAQGRRLGNGVQCLAMGWGLLGRNRGIASVLQELN
           120       130       140       150       160       170

       340       350       360       370       380       390
sp|P05 LDVVPQMTCKQIYRKERRLPRGIIEGQFCAGYLPGGRDTCQGDSGGPIHALLPEYNCVAF
       . :: .. :..   .   : ::   :            .: ::::.:.        : ..
sp|P08 VTVVTSL-CRR--SNVCTLVRGRQAG------------VCFGDSGSPL-------VCNGL
           180        190                   200              210

       400        410       420       430
sp|P05 VVGITSFGKF-CAAPNAPGVYTRLYSYLDWIEKI
       . ::.:: .  ::.   : ... . ....::..:
sp|P08 IHGIASFVRGGCASGLYPDAFAPVAQFVNWIDSI
             220       230       240

  27.3% identity in 22 aa overlap; score:   50 E(10,000): 1.4e+03

      120       130       140
sp|P05 VHGTRIDICTHRNNVPVICCPL
       :.: .  .:   .. :..:  :
sp|P08 VRGRQAGVCFGDSGSPLVCNGL
     190       200       210
```

Figure 8-11: Output of Lalign.

If you play with the Lalign parameters, you may find that stronger gap-opening or gap-extension penalties may split the Dotlet diagonals into two alignments. Lower gap-opening or gap-extension penalties, on the other hand, can cause diagonals to fuse. What happens here depends on the length of the gaps that Lalign needs to insert in the local alignments.

WARNING!

When it comes to judging the quality of your alignment, size definitely matters. It isn't uncommon to see short segments with very good scores. This can result from a variety of reasons — such as low-complexity segments, small-scale regularity in protein sequences, recurrent patterns of hydrophobic/hydrophilic residues, and so on. For instance, in Figure 8-11, the second and the third local alignments are obviously meaningless because they are too short. However, their E-values are better than that of the top alignment. This clearly illustrates that a short alignment can easily get a good score by chance.

REMEMBER

Bioinformatics programs aren't perfect: They can make mistakes and deliver wrong results. Unfortunately, right and wrong obey the same practical rules in bioinformatics as they do in life: They are not absolute concepts.

The top alignment that Lalign reports here illustrates this limitation well. Because of the dot-plot information, we trust this alignment to be correct structures, see Chapter 11.)

Making Global Alignments over the Internet

A global alignment is just what the name implies: an alignment of every amino acid or nucleotide found in your sequences. When you decide to make a global alignment, you implicitly make the hypothesis that your two sequences are related over their entire lengths.

Global alignments are very important for making multiple sequence alignments. However, when done using only two sequences, they offer little interest for biological analysis. Whenever you have two closely related sequences, you can generate a much more informative result by gathering a few more related sequences and making a multiple sequence alignment. (Chapter 9 tells you everything you need to know for this purpose.)

Global alignments aren't useful at all for discovering similarities between two sequences because the statistical method for evaluating E-values doesn't apply to them. Even so, beginners often prefer global alignments to local because they're simpler to understand.

In a global alignment, no amino acid or nucleotide mysteriously disappears: You find in the output exactly what you put in the box, plus a few extra gaps. There are three main reasons for making a global alignment:

- **Checking minor differences between two sequences.** This may happen with data that you've manipulated and possibly altered. The global alignment is the best way to localize potential problems.
- **Analyzing polymorphisms (for example, SNPs) between closely related sequences.**
- **Comparing two sequences that partly overlap.** In that case, you want to make a global pairwise comparison that doesn't penalize misalignments at the extremities of the sequences.

In terms of algorithms, there is not much difference between a local and a global alignment. If you were to write the programs for both, you would not find more than one line of difference between these two algorithms!

You can use Lalign to produce global alignments, but there are other choices — some of which we've listed in Table 8-4, at the end of this chapter.

Using Lalign to produce a global alignment is exactly like using Lalign to produce a local alignment. The only thing extra you need to do is to is to click the radio button that says *Global without End-Gap Penalty,* near the top of the Lalign form at

```
ww.ch.embnet.org/software/LALIGN_form.html
```

Aligning Proteins and DNA

The alignment tools we describe above are only truly effective when you want to compare sequences of a similar type — proteins with proteins or DNA with DNA. Sometimes, however, you need to compare a protein with a piece of DNA (its original gene, for instance). Doing such an alignment requires special tools that can insert the long gaps corresponding to the introns. We know at least two places where you can do this online. One is at the Pasteur Institute (`bioweb.pasteur.fr/seqanal/interfaces/protal2dna.html`) and the other is maintained by Dr. Peer Bork and his group at the European Molecular Biology Laboratory in Heidelberg (`coot.embl.de/pal2nal`). Both of these servers will require you to have your protein and the corresponding DNA sequence ready before you can use them.

If you only have the protein, you can either run a blastx against a complete genome (see Chapter 6) or use a Web server called Protogene (available at `www.tcoffee.org`). Protogene automatically fetches the DNA sequence corresponding to a given protein.

Free Pairwise Sequence Comparisons over the Internet

Many existing servers propose pairwise comparisons. Table 8-4 lists a brief selection of some of the more stable resources.

Lalign	fasta.bioch.virginia.edu/ fasta_www/plalign.htm	Global/Local
USC	www-hto.usc.edu/software/ seqaln/seqaln-query.html	Global/Local/
Alion	fold.stanford.edu/alion/	Global/Local
Align	genome.cs.mtu.edu/ align.html	Global/Local
Align	www.ebi.ac.uk/emboss/align/	Global/Local
xenAliTwo	www.soe.ucsc.edu/~kent/ xenoAli/xenAliTwo.html	Local for DNA
Blast2seqs	www.ncbi.nlm.nih.gov/ blast/bl2seq/wblast2.cgi	Local BLAST
Protal2dna	bioweb.pasteur.fr/seqanal/ interfaces/protal2dna.html	Protein against DNA
Pal2nal	coot.embl.de/pal2nal	Protein against DNA

Generating an alignment isn't always enough. You may also need to visualize this alignment and to evaluate its statistical significance. Table 8-5 lists a few sites that enable you to do this online.

Table 8-5 Online Pairwise Alignments Analyses

Name	*Address*	*Function*
lalnview	www.expasy.ch/tools/ sim-prot.html	Visualization
prss	www.ch.embnet.org/software/ PRSS_form.html	Evaluation
prss	Fasta.bioch.virginia.edu/ fasta/prss.htm	Evaluation
graph-align	darwin.nmsu.edu/cgi-bin/ graph_align.cgi	Evaluation

Building a Multiple Sequence Alignment

In This Chapter

- Gathering the sequences you need to make a multiple sequence alignment
- Making a multiple sequence alignment with ClustalW and MUSCLE
- Creating and comparing multiple sequence alignments with Tcoffee
- Comparing sequences you cannot align

> *The man with two watches never knows the time, and the man with one watch only thinks he knows.*
>
> — A man with multiple watches

In this chapter, we show you how to compare many protein or nucleotide sequences simultaneously. This is something you probably want to do if you have a sequence and you want to elucidate the role of each amino acid or nucleotide that it contains.

To compare multiple sequences, your first job is to select a few sequences and then align them all. In this chapter, we show you how to select the most appropriate sequences and how you can generate such a *multiple sequence alignment*. We introduce three different multiple-sequence-alignment methods that should cover most of your needs: ClustalW because everybody uses it, MUSCLE because it is very fast, Tcoffee because it is very accurate, and can let you combine sequences and structures. Since it's too easy to generate bad alignments that look good, we also show you how to evaluate the quality of your alignment.

Sometimes your multiple sequence alignment isn't good enough to be useful. When this happens, we show you alternatives to sequence alignment — and that it's possible to compare your sequences without aligning them. We introduce two of these methods: the Gibbs sampler (for identifying related segments of the same length) and Pratt, a motif-discovery method for discovering PROSITE patterns.

the first section of this chapter, “Finding Out if a Multiple Sequence Alignment Can Help You.”

In any case, remember that multiple alignments are useful for predicting protein structures (see Chapter 11), central for predicting the function of proteins, and indispensable for phylogenetic analysis (see Chapter 13). Of course, the better your multiple sequence alignment, the better use you can make of it: better structural models, better functional predictions, and better phylogenetic trees.

Even so, building multiple sequence alignments is far from an exact science. In fact, it’s more art than science, requiring that you use everything you know in bioinformatics and in biology. This chapter is here to give you the secrets of the trade so that you, too, will be able to create the multiple sequence alignment that best suits your needs — and even find your own recipes.

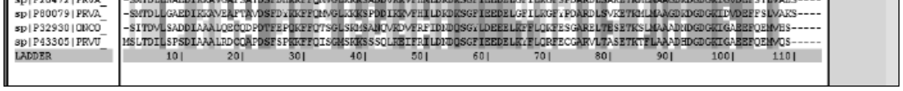
If you already have an alignment and you only want to modify it or improve its appearance for publication (highlighting, shading, and so on), go directly to Chapter 10 — that’s the chapter entirely dedicated to such fine-tuning techniques.

Finding Out if a Multiple Sequence Alignment Can Help You

Figure 9-1 shows you what a multiple alignment actually looks like: As you can see, it’s really only a matter of rewriting your sequences so similar features end up in the same columns. That’s it, plain and simple.

This kind of multiple alignment is ideal if you want to study a sequence family where all sequences share the *same common ancestor.* Don’t worry if you have only one member at this stage of the game; we show you later in this chapter how to organize a family gathering!

sequence
alignment.



```
sp|P80079|PRVA_  -SMTDLLGAEDIKKAVEAFTAVDSFDYKKFFQMVGLKKKSPDDIKKVFHILDKDKSGFIEEDELGFILKGFYPDARDLSVKETKMLMAAGDKDGDGKIDVDEFFSLVAKS----
sp|P32930|ONCO_  -SITDVLSADDIAAALQECQDPDTFEPQKFFQTSGLSKMSANQVKDVFRFIDNDQSGYLDEEELKFFLQKFESGARELTESETKSLMAAADNDGDGKIGAEEFQEMVHS-----
sp|P43305|PRVU_  MSLTDILSPSDIAAALRDCQAPDSFSPKKFFQISGMSKKSSSQLKEIFRILDNDQSGFIEEDELKYFLQRFECGARVLTASETKTFLAAADHDGDGKIGAEEFQEMVQS-----
LADDER                10|       20|       30|       40|       50|       60|       70|       80|       90|      100|      110|
```

Identifying situations where multiple alignments do not help

Part of knowing when to use multiple sequence alignments involves knowing when *not* to use them. The multiple-sequence methods we describe in this chapter don't work well for assembling the sequence pieces in a sequencing project. If you have a set of *shotgun sequences* — short, partly overlapping sequences — or if you want to turn an EST cluster into a gene sequence, multiple sequence alignment doesn't work well — and dumping your sequences into ClustalW, MUSCLE, or Tcoffee can produce a very disappointing and uninformative result.

If you're facing this particular type of problem, you may want to use specialized sequence assembly tools, such as Phred and Phrap (`www.phrap.org`) or cap3 (`www.mainlab.clemson.edu/cgi-bin/gdr/gdr_cap3`). Unfortunately, the intricacies of specialized sequence-assembly tools (which we briefly overview in Chapter 5) are far beyond the scope of this book.

Another situation that *doesn't* call for multiple sequence alignment is when the sequence you're interested in has no homologue in any of the sequence databases. If this happens, you're out of luck; there's no way you can build a multiple alignment that relates to this sequence. You may try to find a few sequences by using functional criteria and conducting a pattern search with Pratt, but don't count on any miracles.

Helping your research with multiple sequence alignments

With the appropriate sequences in hand, you can build the proper multiple sequence alignment. To be honest, you could build this alignment manually, rather than by using a computer. Manual alignments are like home-cooked food: When prepared by a good cook, nothing beats them; for regular folks like us, however, the automatic processing (TV dinner) is generally much better.

Table 9-1 Main Criteria for Building a Multiple Sequence Alignment

Criterion	*Meaning*
Structural similarity	Amino acids that play the same role in each structure are in the same column. Structure-superposition programs are the only ones that use this criterion.
Evolutionary similarity	Amino acids or nucleo tides related to the same amino acid (or nucleotide) in the common ancestor of all the sequences are put in the same column. No automatic program explicitly uses this criterion, but they all try to deliver an alignment that respects it.
Functional similarity	Amino acids or nucleotides with the same function are in the same column. No automatic program explicitly uses this criterion, but if the information is available, you can force some programs to respect it — or you can edit your alignment manually.
Sequence similarity	Amino acids in the same column are those that yield an alignment with maximum similarity. Most programs use sequence similarity because it is the easiest criterion. When the sequences are closely related, their structural, evolutionary, and functional similarities are equivalent to sequence similarity.

The first three criteria have a clear biological meaning — the fourth one doesn't. Yet, when the sequences are similar enough, you can use similarity to produce a multiple alignment that reflects the evolutionary, the structural, and the functional relationships that exist between your sequences. Of course, to do this, you need an alignment that you can trust. Later in this chapter, we show you how to make sure you're putting your money on the right alignment.

Table 9-2 lists the main applications of multiple sequence alignments.

	family. Alignments that include Swiss-Prot sequences are the most informative. Use the ExPASyBLAST server (at `www.expasy.ch/tools/blast/`) to gather and align them.
Phylogenetic analysis	If you carefully choose the sequences you include in your multiple alignment, you can reconstruct the history of these proteins. Use the Pasteur Phylip server at `bioweb.pasteur.fr/seqanal/phylogeny/phylip-uk.html`.
Pattern identification	By discovering very conserved positions, you can identify a region that is characteristic of a function (in proteins or in nucleic-acid sequences). Use the logo server for that purpose: `www-lmmb.ncifcrf.gov/~toms/sequencelogo.html`.
Domain identification	It is possible to turn a multiple sequence alignment into a profile that describes a protein family or a protein domain (PSSM). You can use this profile to scan databases for new members of the family. Use NCBI-BLAST to produce and analyze PSSMs: `www.ncbi.nlm.nih.gov/blast/blastcgihelp.shtml#pssm`.
DNA regulatory elements	You can turn a DNA multiple alignment of a binding site into a weight matrix and scan other DNA sequences for potentially similar binding sites. Use the Gibbs sampler to identify these sites: `bayesweb.wadsworth.org/gibbs/gibbs.html`
Structure prediction	A good multiple alignment can give you an almost perfect prediction of your protein secondary structure for both proteins and RNA. Sometimes it can also help in the building of a 3-D model.
nsSNP analysis	Various gene alleles often have different amino-acid sequences. Multiple alignments can help you predict whether a Non-Synonymous Single-Nucleotide Polymorphism is likely to be harmful. See the SIFT site for more details: `blocks.fhcrc.org/sift/SIFT.html`.
PCR analysis	A good multiple alignment can help you identify the less-degenerated portions of a protein family, in order to fish out new members by PCR (polymerase chain reaction). If this is what you want to do, you can use the following site: `blocks.fhcrc.org/codehop.html`.

instance, active sites of enzymes are much conserved.

- Less-important residues change more easily — sometimes randomly — and sometimes in order to adapt a function.

For you, this has a very simple consequence: When you look at a multiple alignment, you can make the hypothesis that *conserved* positions (columns where all the sequences contain the same amino acid or nucleotide) are more important for the function than *non conserved* positions (columns where the sequences contain different amino acids or nucleotides). Of course, you could tell all this from the alignment of just two sequences — but using more sequences makes it easier to discriminate between important and less-important positions.

And one more thing: A multiple sequence alignment is a terrific way to present your results. It lets you put lots of information into a single model and makes it incredibly easy to spot inconsistencies or potential problems. If you want to get an important point across — in a paper or in front of an audience — a good, hand-crafted, well-colored multiple alignment says more than 3 billion nucleotides in bulk.

All this being said, multiple alignments do NOT cure asthma, arthritis, lumbago, or baldness. They won't make you stronger or increase any of your potentials (except maybe your research potential). But at this point, if you need more arguments to convince you of the genius of multiple alignments, chances are you don't really need to make this kind of alignment.

Choosing the Right Sequences

Anyone who ever worked in a lab knows that molecular biology is very much like cooking: It's all about selecting the right ingredients and putting them into the pot at the right time and in the right order.

Building a multiple alignment obeys the same rule. Before you build your alignment, you must carefully select the sequences you want to align. These sequences are members of the same protein family, and they all share a common ancestor. The family is usually too large to be entirely included in your multiple alignment, and picking the right sequences is an art. If you want to be good at this game, you need to know what you want to show with your alignment — and you need to know how the multiple alignment programs work.

The kinds of sequences you're looking for

Let's assume that you start this procedure with your favorite sequence. You want to thoroughly study this sequence and you know that in order to do this you have to build a multiple alignment. Table 9-3 summarizes most of the things you must take into account when selecting these extra sequences.

Table 9-3 A Few Guidelines for Selecting Sequences

Problem	***Diagnostic***
Proteins or DNA	Use proteins whenever possible. You can turn them back into DNA *after* doing the multiple alignment.
Many sequences	Start with 10–15 sequences; avoid aligning more than 50 sequences.
Very different sequences	Sequences that are less than 30 percent identical to more than half the other sequences in the set often cause troubles.
Identical sequences	They never help. Unless you have a *very* good reason to do so, avoid incorporating into your multiple alignment any sequence that's more than 90 percent identical to another sequence in the set.
Partial sequences	Multiple-sequence-alignment programs prefer sequences that are roughly the same length. Programs often have difficulties comparing items in a mixture of complete sequences and shorter fragments.
Repeated domains	Sequences with repeated domains cause trouble for most multiple-alignment programs — especially if the number of domains is different. When this happens, you may be better off extracting the domains yourself with Dotlet or Lalign (see Chapter 8) and making a multiple alignment of those segments.

alignment method, such as the Gibbs sampler, or a pattern extraction motif, such as Pratt. (See "Comparing Sequences That You Can't Align," later in this chapter, for a description of these two methods.)

Multiple-sequence-alignment methods are at their best when aligning protein sequences. The reason is that protein sequences are three times shorter than the corresponding DNA, and they use a more informative alphabet of 20 amino acids.

If you want to persist in carrying out a phylogenetic analysis on a set of coding DNA sequences, things work better if you do the following:

1. Translate your DNA sequences into proteins.
2. Perform the multiple alignments on the proteins.
3. Thread the DNA back onto the protein multiple sequence alignment framework using pal2nal (`coot.embl.de/pal2nal`) or Protogene if you do not have the original DNA sequence (`www.tcoffee.org`).

If your proteins are difficult to align because they have few similarities, DNA information does not help. This has to do with the degeneracy of the genetic code and the fact that there are 20 amino acids and only 4 nucleotides. If there is little signal at the protein level, you can be sure that there is NO useful signal at the DNA level.

Choosing the right number of sequences

Of course, there's no absolute answer to this question, such as *42* or *7*. A few years ago, the answer was easy: Get everything you can and go to the lab if there aren't enough sequences in the databases! But that isn't true anymore. These days — given the sizes of the databases and new complete genome sequences flowing in twice a month — you may easily find hundreds or thousands of sequences that would be suitable for inclusion in your multiple sequence alignment. But that doesn't mean you have to use them all!

In our opinion, you should start with a relatively small number of sequences — between 10 and 15 sequences would be suitable for most cases. After you get something interesting happening with this small set, you can always increase its size. In any case, it's hard to see any reason for generating a multiple alignment with more than 50 sequences, unless you're interested in building some extensive phylogenetic tree.

If you start with hundreds of sequences, you immediately hit troubles. There are good reasons why:

- ***Displaying* big alignments is difficult.** You can’t print them, and they clog your computer when you want to visualize them. If columns are longer than one page, interpretation becomes impossible.
- ***Using* big alignments is difficult.** Tree-building and structure-prediction programs cannot handle them easily.
- **Making *accurate* big alignments is difficult.** You want your multiple sequence alignment to be highly reliable so you can be confident that all the sequences it contains are true members of the family. A major cause for concern is that multiple-sequence-alignment programs make mistakes. The curse is that these mistakes do not add up, they *multiply* — making it easy for a tiny number of bad sequences to ruin an entire alignment. Of course, the more sequences you have, the more likely this is to happen. The best way to avoid such a disaster is to start small — and gradually increase the size of your multiple sequence alignment until it contains all the sequences you’re interested in.

Now that you know how many sequences you need, the last question you face is deciding how related these sequences must be. Should you choose sequences that are very similar or very different?

Making the right compromise between similarity and new information

If you think that very similar sequences give very good alignments, you’re right! However, a multiple sequence alignment that’s correct isn’t enough; it must also be useful.

For instance, an alignment that only contains very similar sequences brings little information. You can use it to extrapolate annotations, but you can’t do phylogeny, structure prediction, or any of the other useful applications that we list in Table 9-2. These other tasks require being able to observe mutation patterns in every column — which isn’t possible if you have an alignment in which most columns are entirely conserved.

Grabbing the most distantly related sequences you can find doesn’t work, either. Multiple-sequence-alignment programs can’t use sequences that are too different — even if these sequences are homologous. In fact, two things multiple-sequence-alignment programs *really* don’t like are

- Sequences that are very different from every other sequence in the group
- Sequences that need long insertions/deletions to be properly aligned

be as distantly related as possible — without requiring too many gaps in order to be properly aligned. These two criteria are mutually exclusive, so finding the right trade-off requires a bit of strategy. The following steps show you a general approach that you should have in mind when gathering your sequences:

1. **Select a few sequences.**

   See the section “Gathering your sequences with online BLAST servers,” later in this chapter.

2. **Compute a multiple alignment by using one of the servers we introduce in this chapter.**

   See the section “Choosing the Right Method of Multiple Sequence Alignment,” later in this chapter.

3. **Evaluate the quality of your alignment visually.**

   See the section “Interpreting Your Multiple Sequence Alignment,” later in this chapter.

4. **If your alignment looks good, keep the sequences.**

   A good alignment contains nicely conserved blocks separated by regions with insertions and deletions. If you have such an alignment, your sequence set is probably appropriate — and you can try to extend it by adding a few new sequences.

5. **If your alignment is difficult to interpret,**

   a. Examine the sequences more closely — try to remove the troublemakers that are the most distantly related, or those that cause long insertions/deletions.

   b. Redo the alignment with the smaller set.

   c. Keep trimming the set until you get a multiple alignment that’s easier to interpret.

If you can, make sure that *each sequence is between 30 and 70 percent identical with more than half of the sequences in the set.* This way, you’re making a reasonable trade-off between new information and alignment quality.

Before adding a sequence to a multiple alignment, you can try to figure out whether it’s a good choice by making pairwise comparisons with some of the tools we describe in Chapter 6. (*Pairwise* comparisons let you compare sequences two by two.) However, we don’t recommend that you do so exhaustively; it’s time-consuming.

- Never use white spaces in your sequence names.
- Do not use special symbols. Stick to plain letters, numbers, and the underscore (_) to replace spaces. Avoid ALL other symbols, especially those that are the most tempting for special sequences (such as @, #, _,^ and so on).
- Never use names longer than 15 characters.
- Never give the same name to two different sequences in your set. Although some programs accept it, others (such as ClustalW) don't.

If you don't obey these naming rules, some multiple-sequence-alignment programs may automatically change the name of your sequences, without the courtesy of telling you.

Gathering your sequences with online BLAST servers

There are two types of sequences that you may want to integrate into your multiple alignment:

- **Characterized sequences:** These are sequences for which you have good annotations and experimental information. You'd definitely want to include these sequences in your alignment because they bring biological information with them — and also allow feature propagation.
- **Uncharacterized sequences:** This category can include your sequence(s) of interest as well as database sequences. Uncharacterized sequences must be members of the same family. Your main motivation in including them in your multiple alignment is to distinguish between the conserved positions that cannot mutate and the other, less-important columns. They help in getting some contrast on your sequence of interest.

In this section, we show you how to gather these sequences with the BLAST database search program. If you want to know all the gory details about BLAST, check out Chapter 7. Here we give you only the bare minimum to get by.

With BLAST, you can search databases for sequences that are homologous (similar) to a query. The original query can be any sequence you are interested in — protein or DNA — and you can use BLAST to search both protein and DNA databases. The main reason for using BLAST is to identify database in. Table 9-4 lists three BLAST servers that are useful for generating a list of sequences in FASTA format (the best format for moving your sequences around, from one program to another) or for sending chosen sequences to a multiple-alignment server.

Table 9-4 BLAST Servers Integrating Multiple-Alignment Methods

Address	*What You Can Do There*
`www.expasy.ch/tools/blast/`	Extract entire sequences, export sequences in FASTA, submit sequences to ClustalW, Tcoffee or MAFFT. Turn the list of Hits into a non-redundant collection of sequences.
`npsa-pbil.ibcp.fr/cgi-bin/npsa_automat.pl?page=npsa_blast.html`	Extract entire sequences, extract sequences fragments, export sequences in FASTA, submit sequences to ClustalW.
`srs.ebi.ac.uk`	Submit sequences to ClustalW.

In the following steps list, we show you how to use the ExPASy server. The SRS server, although sometimes useful, is a bit harder to use than the other two listed in Table 9-4.

Selecting sequences on the ExPASy server

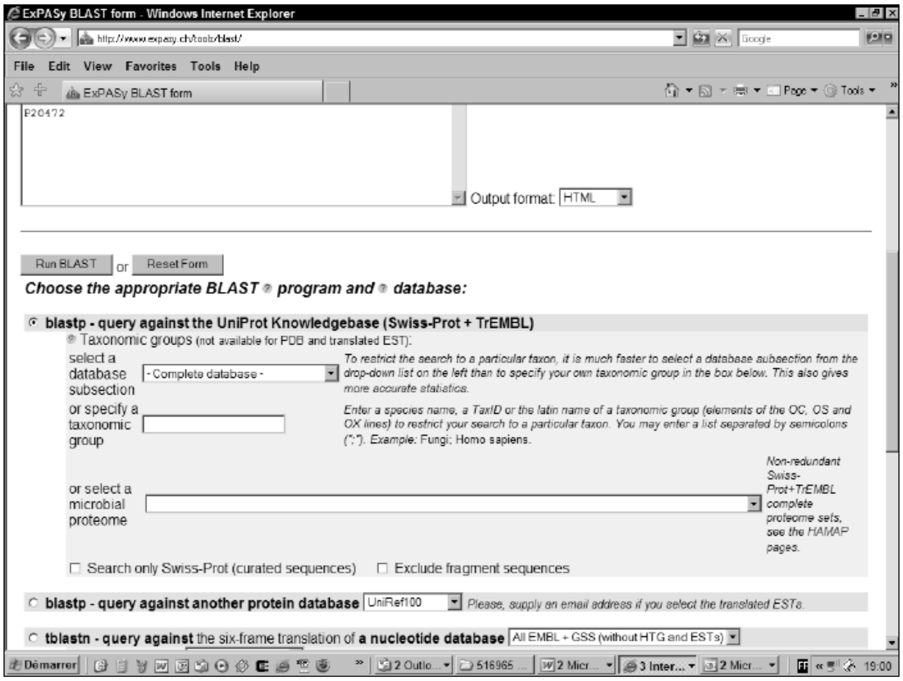
In the following steps list, we select appropriate sequences to make a multiple sequence alignment of calcium-dependent kinase proteins.

You can use this server only to retrieve *protein* sequences. If you're interested in gathering *DNA* sequences, use the European Bioinformatics SRS server (`srs.ebi.ac.uk`) instead.

The BLAST server of the PBIL site is very similar to ExPASy. If you don't find the database you're interested in on the ExPASy server, try the PBIL.

1. **Point your browser to** `www.expasy.ch/tools/blast/`**.**

   The BLAST page of the ExPASy server appears.


ExPASy BLAST form - Windows Internet Explorer
http://www.expasy.ch/tools/blast/
Google
File Edit View Favorites Tools Help
ExPASy BLAST form
Page
Tools
Run BLAST
Reset Form
Démarrer
2 Outlo...
516965 ...
2 Micr...
3 Inter...
2 Micr...
19:00

P20472

Output format: HTML

Run BLAST or Reset Form

Choose the appropriate BLAST program and database:

● **blastp - query against the UniProt Knowledgebase (Swiss-Prot + TrEMBL)**

Taxonomic groups (not available for PDB and translated EST):

select a database subsection: - Complete database -

To restrict the search to a particular taxon, it is much faster to select a database subsection from the drop-down list on the left than to specify your own taxonomic group in the box below. This also gives more accurate statistics.

or specify a taxonomic group:

Enter a species name, a TaxID or the latin name of a taxonomic group (elements of the OC, OS and OX lines) to restrict your search to a particular taxon. You may enter a list separated by semicolons (";"). Example: Fungi; Homo sapiens.

or select a microbial proteome:

Non-redundant Swiss-Prot+TrEMBL complete proteome sets, see the HAMAP pages.

□ Search only Swiss-Prot (curated sequences) □ Exclude fragment sequences

○ **blastp - query against another protein database** UniRef100 *Please, supply an email address if you select the translated ESTs.*

○ **tblastn - query against** the six-frame translation of **a nucleotide database** All EMBL + GSS (without HTG and ESTs)

Figure 9-2: Using the BLAST ExPASy server to gather sequences.

3. **Select the BLAST flavor that you're interested in.**

   If you gave a protein sequence in Step 2, select blastp.

   If you gave a coding DNA sequence in Step 2, select tblastn.

4. **Keep the default option — Complete Database — in the pull-down menu.**

   This amounts to simultaneously searching Swiss-Prot + TrEMBL + TrEMBL_NEW. If the search reports too many sequences that are very similar to your sequence of interest, you can decrease the number of identical hits by selecting a smaller database from the Database pull-down menu — Swiss-Prot, for example, or the database of only the microbial proteomes.

   For a multiple sequence alignment, do not select Translated ESTs from the pull-down menu. These sequences are mostly incomplete protein sequences that may confuse the multiple-alignment procedure.

Show option to 1000.

This choice makes it possible to judge the quality of the alignment before selecting a sequence.

7. **Click the Run BLAST button.**

   After a brief pause, a Results page appears.

8. **Scroll down the page to select the sequences you want.**

   You select a sequence by checking the box to its left.

   This is the most delicate part of the process. There is no absolute rule to selecting your sequences, but you can use the following guidelines:

   - **Select the top sequence.** This top sequence is usually your sequence of interest. If your sequence of interest is not at the top, you may have to add it to the list later on.
   - **For a first analysis, you want to select ten sequences or fewer.** Ideally, the ten sequences to select should be evenly spaced between the very good E-values (10^{-40}) and less-good E-values (10^{-5}). Figure 9-3 gives you an idea of what such a selection looks like.

     For the purpose of this steps list, choose (from top to bottom): P20472, P80079, P02626, P02619, P43305, P32930, Q91482, P02620, P02622, P02627.
   - **Before selecting a sequence, check to make sure it's similar to the query sequence — along its entire length.**

     The alignment section is at the bottom of the BLAST output. You must be especially careful with hits that have E-values higher than 10^{-10}. They are equally likely to correspond to a good partial match, a global overall match, or a match between a protein fragment and your sequence. Inspecting the alignment is the only way to distinguish between these situations.

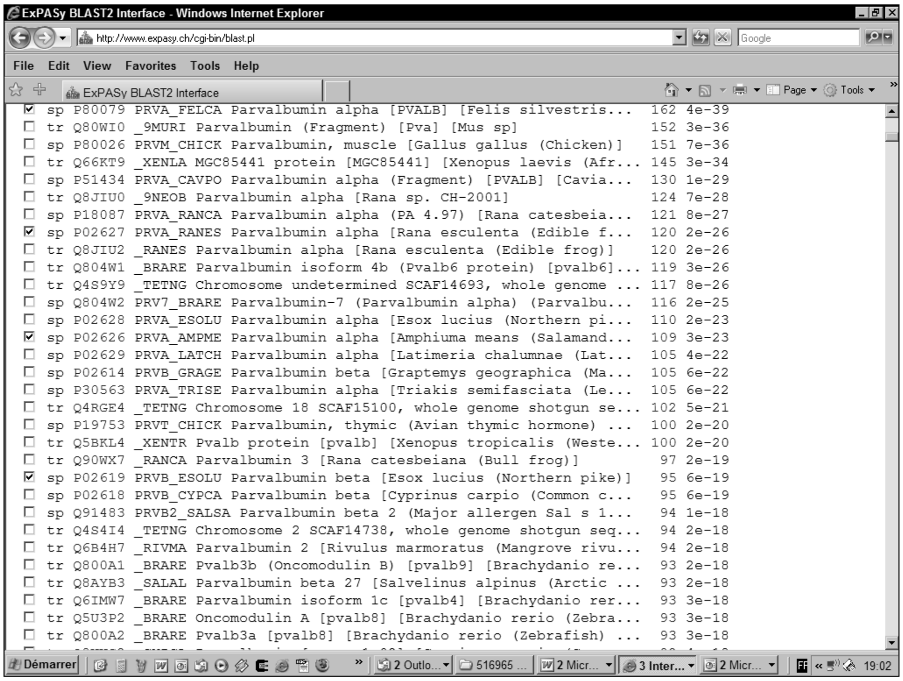
9. **Choose the method you want to use to export your sequences from the Send Selected Sequences pull-down menu, as shown in Figure 9-4.**

   There are several ways to export your sequences:

   - **FASTA:** Generates a file that contains your sequences in FASTA format. You can save this file with the File➪Save As option of your browser. When you need to, you can reopen this file with your browser, in order to cut and paste its content into another server form (MUSCLE for instance, at `www.drive5.com/muscle/`).

sequences and you don't know how to choose.

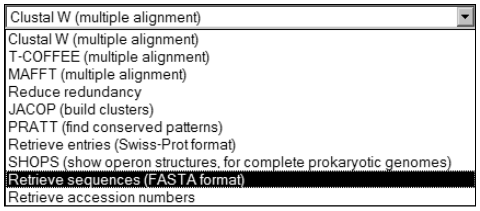
- **Pratt:** Will search for conserved motifs in your sequences without aligning them.



ExPASy BLAST2 Interface - Windows Internet Explorer
http://www.expasy.ch/cgi-bin/blast.pl
Google
File Edit View Favorites Tools Help
ExPASy BLAST2 Interface
Page
Tools
Démarrer
2 Outlo...
516965 ...
2 Micr...
3 Inter...
2 Micr...
19:02

```
☑ sp P80079 PRVA_FELCA Parvalbumin alpha [PVALB] [Felis silvestris...  162 4e-39
☐ tr Q80WI0 _9MURI Parvalbumin (Fragment) [Pva] [Mus sp]                152 3e-36
☐ sp P80026 PRVM_CHICK Parvalbumin, muscle [Gallus gallus (Chicken)]   151 7e-36
☐ tr Q66KT9 _XENLA MGC85441 protein [MGC85441] [Xenopus laevis (Afr... 145 3e-34
☐ sp P51434 PRVA_CAVPO Parvalbumin alpha (Fragment) [PVALB] [Cavia...  130 1e-29
☐ tr Q8JIU0 _9NEOB Parvalbumin alpha [Rana sp. CH-2001]                124 7e-28
☐ sp P18087 PRVA_RANCA Parvalbumin alpha (PA 4.97) [Rana catesbeia...  121 8e-27
☑ sp P02627 PRVA_RANES Parvalbumin alpha [Rana esculenta (Edible f...  120 2e-26
☐ tr Q8JIU2 _RANES Parvalbumin alpha [Rana esculenta (Edible frog)]    120 2e-26
☐ tr Q804W1 _BRARE Parvalbumin isoform 4b (Pvalb6 protein) [pvalb6]... 119 3e-26
☐ tr Q4S9Y9 _TETNG Chromosome undetermined SCAF14693, whole genome ... 117 8e-26
☐ sp Q804W2 PRV7_BRARE Parvalbumin-7 (Parvalbumin alpha) (Parvalbu...  116 2e-25
☐ sp P02628 PRVA_ESOLU Parvalbumin alpha [Esox lucius (Northern pi...  110 2e-23
☑ sp P02626 PRVA_AMPME Parvalbumin alpha [Amphiuma means (Salamand...  109 3e-23
☐ sp P02629 PRVA_LATCH Parvalbumin alpha [Latimeria chalumnae (Lat...  105 4e-22
☐ sp P02614 PRVB_GRAGE Parvalbumin beta [Graptemys geographica (Ma...  105 6e-22
☐ sp P30563 PRVA_TRISE Parvalbumin alpha [Triakis semifasciata (Le...  105 6e-22
☐ tr Q4RGE4 _TETNG Chromosome 18 SCAF15100, whole genome shotgun se... 102 5e-21
☐ sp P19753 PRVT_CHICK Parvalbumin, thymic (Avian thymic hormone) ...  100 2e-20
☐ tr Q5BKL4 _XENTR Pvalb protein [pvalb] [Xenopus tropicalis (Weste... 100 2e-20
☐ tr Q90WX7 _RANCA Parvalbumin 3 [Rana catesbeiana (Bull frog)]         97 2e-19
☑ sp P02619 PRVB_ESOLU Parvalbumin beta [Esox lucius (Northern pike)]   95 6e-19
☐ sp P02618 PRVB_CYPCA Parvalbumin beta [Cyprinus carpio (Common c...   95 6e-19
☐ sp Q91483 PRVB2_SALSA Parvalbumin beta 2 (Major allergen Sal s 1...   94 1e-18
☐ tr Q4S4I4 _TETNG Chromosome 2 SCAF14738, whole genome shotgun seq...  94 2e-18
☐ tr Q6B4H7 _RIVMA Parvalbumin 2 [Rivulus marmoratus (Mangrove rivu...  94 2e-18
☐ tr Q800A1 _BRARE Pvalb3b (Oncomodulin B) [pvalb9] [Brachydanio re...  93 2e-18
☐ tr Q8AYB3 _SALAL Parvalbumin beta 27 [Salvelinus alpinus (Arctic ...  93 2e-18
☐ tr Q6IMW7 _BRARE Parvalbumin isoform 1c [pvalb4] [Brachydanio rer...  93 3e-18
☐ tr Q5U3P2 _BRARE Oncomodulin A [pvalb8] [Brachydanio rerio (Zebra...  93 3e-18
☐ tr Q800A2 _BRARE Pvalb3a [pvalb8] [Brachydanio rerio (Zebrafish) ...  93 3e-18
```

Figure 9-3: Selecting sequences on the BLAST output.



Clustal W (multiple alignment)
Clustal W (multiple alignment)
T-COFFEE (multiple alignment)
MAFFT (multiple alignment)
Reduce redundancy
JACOP (build clusters)
PRATT (find conserved patterns)
Retrieve entries (Swiss-Prot format)
SHOPS (show operon structures, for complete prokaryotic genomes)
Retrieve sequences (FASTA format)
Retrieve accession numbers

Figure 9-4: Output selector on the ExPASy BLAST page.

This can be handy if you're using a BLAST server that does not have sequence-extraction facilities. You can simply copy the accession number of the sequences you are interested in, and extract them here.

This procedure works only if the sequences that you're interested in belong to the Swiss-Prot or TrEMBL databases. The following steps list shows you how to use this server:

1. **Point your browser to** `www.expasy.ch/sprot/sprot-retrieve-list.html`**.**

   The Swiss-Prot/TrEMBL Retrieve a List of Entries page appears.

2. **Leave the File Name field blank so the browser returns the sequences directly.**

3. **On the Format line, select the FASTA check box.**

4. **Enter the accession numbers of your sequences in the Sequence window, as shown in Figure 9-5.**

   Enter one accession number (or sequence name) per line. For our example, we entered P20472, P80079, P02626, P02619, P43305, P32930, Q91482, P02620, P02622.

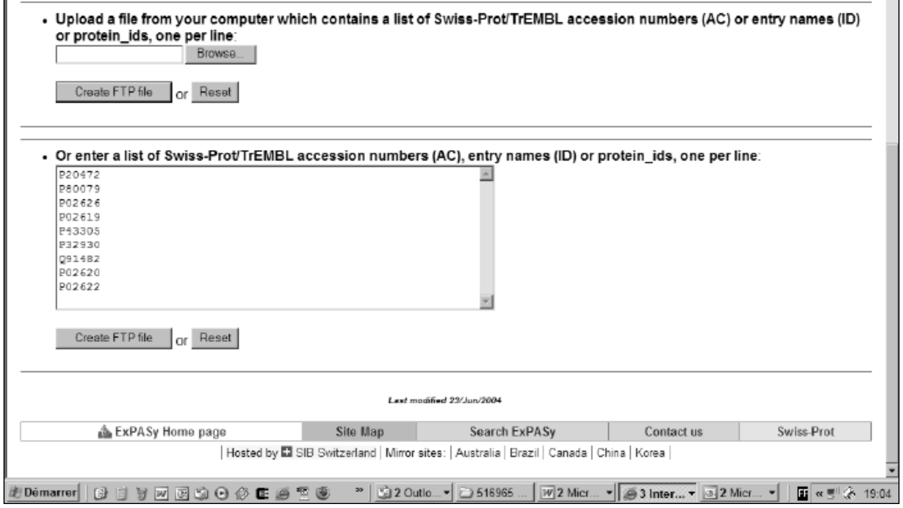
   The Upload a File field is convenient if you have all your accession numbers in a file. You can generate this file with Microsoft Word, but it must be in text form. You can enter the filename in the box or use the Browse button and choose your file for upload.

5. **Underneath the field where you entered your accession numbers, click the Create FTP File button.**

   This submits your sequence request to the ExPASy server.

6. **Save the results into a text file with the File➪Save As option of your browser.**

   This server gives the sequences names that are longer than 15 characters. These can give you trouble on some servers.



- Upload a file from your computer which contains a list of Swiss-Prot/TrEMBL accession numbers (AC) or entry names (ID) or protein_ids, one per line:

- Or enter a list of Swiss-Prot/TrEMBL accession numbers (AC), entry names (ID) or protein_ids, one per line:

```
P20472
P80079
P02626
P02619
P43305
P32930
Q91482
P02620
P02622
```

Last modified 23/Jun/2004

| Hosted by SIB Switzerland | Mirror sites: | Australia | Brazil | Canada | China | Korea |

Figure 9-5: Gathering several sequences from the ExPASy server.

Choosing the Right Method of Multiple Sequence Alignment

Before you start making multiple sequence alignments, you must know that none of the methods available today is perfect. They all use approximations.

Building a multiple alignment that lets you make a real discovery requires some practice. The usual strategy requires comparing several alternative results and looking for robustness and stability.

In this section, we show you how to use ClustalW, the most commonly used multiple sequence alignment package. It is a simple, no-fuss, no-questions-asked kind of tool. We also show you how to use Tcoffee, one of the latest multiple-sequence-alignment packages that you can use. With Tcoffee, you can combine sequences and structures, evaluate an alignment, or merge several alternative multiple alignments into a single unified result. Finally, we also introduce you to MUSCLE, one of the fastest alignment methods around.

ClustalW uses a progressive method to build its alignments. Instead of aligning all the sequences at the same time, it adds them one by one. If you want to get best results from ClustalW, understanding its underlying principle helps a lot.

Many ClustalW servers are around. They usually run the same version of this program, but their interfaces give you access to different options. At the end of this chapter, we give you a list of servers that run ClustalW. Shopping around to find a ClustalW server that's both fast *and* reliable is worth your time.

Running the EBI ClustalW server

ClustalW is a typical example of a program that can produce a reasonable output with its default settings. You should probably worry about the settings only when you want to change the output format of the program.

ClustalW behind the scenes

ClustalW is the latest of the Clustal software series. Clustal was the first multiple sequence alignment program that could run on almost any platform. Legend has it that Des Higgins made the original design for Clustal on the back of an envelope, in a smoky Dublin pub, in the early 1990s. A few years later, when Des moved to the European Molecular Biology Laboratory in Heidelberg, he took Clustal along — and kept it alive, turning it into one of the most successful series of bioinformatics programs: ClustalV, ClustalW, and ClustalX, an X-window implementation.

Des Higgins did not truly invent the Clustal algorithm; Paula Hogeweg had already described it in the early 1980s. What Des did was to put together all the right ingredients so his program could be used by anybody on almost any computer. He also packaged and arranged everything so it would be very easy to use. These are some of the reasons why Clustal became an instant hit. These days, with more than 35,000 citations, ClustalW is one of the most widely cited scientific publications in the history of biology.

ClustalW uses a progressive algorithm. This means that it builds the alignment progressively, adding sequences little by little until the complete multiple sequence alignment is finished. The reason for doing this is that state-of-the art sequence-alignment programs find it hard to align more than two sequences at a time. Since these pairwise-alignment programs are the only tools we've got on hand to produce a multiple alignment, the only solution is to cheat a little. This is what the progressive algorithm does.

The trick is simple: You start comparing all your sequences two by two, so you can cluster them by similarity (Clustal does this for you). The clustering looks like a phylogenetic tree. (It is the file with a `.dnd` extension that ClustalW outputs.)

```
  |------|
  |      |------B
  |
Root ----------|
  |
  |      |------C
  |------|
         |------D
```

The topology of this dendrogram tells us a simple story: It says that A and B are more similar to each other than they are to C and D. Thus if we align A with B, we are less likely to make a mistake than if we align A with C or D.

To make the progressive alignment, ClustalW follows the dendrogram topology: It starts aligning A with B. After this it aligns C with D. When this is done, Clustal has two small multiple alignments (AB and CD). This is where Clustal pulls out its main trick: It aligns the two alignments as if each of them was a single sequence! It is not as complicated as it seems and there are many ways to do this. For instance, you could replace each alignment with a single consensus sequence. Clustal uses a slightly more sophisticated method, but the idea is essentially the same: It treats multiple alignments like single sequences and aligns them two by two.

Now you may ask, where do we cheat then? The answer is simple: we make a multiple alignment with ALL the sequences in our set, but we do not use ALL the information they contain. For instance, when Clustal aligns A and B, it does not use C and D. This could be a problem. If A and B are very different, we will produce an incorrect pairwise alignment — which would be a waste if the two other sequences contained some useful information we did not use. Imagine, for instance, that A, B, C, and D all contain a certain very short (but important) motif. This motif does not look so important when you

The reason not to use all the information is that it's too expensive in terms of computation. So we cheat a little and use the progressive alignment. This shouldn't worry you too much, though. Even if it is a little greedy and approximate, Clustal often delivers pretty good alignments.

When you ask yourself which type of sequences are best suited for ClustalW, imagine that your sequences are like big stones spread across a shallow river. Making a multiple alignment is like crossing this river by jumping from stone to stone: It doesn't matter how wide the river is, as long as you always find a stone to jump to. Similarly, it doesn't matter how many sequences you have, and how far apart they are from each other, as long as a chain of correct alignments exists that can take you across the entire set. (Of course, if a sequence or a small group of sequences is very different from the rest, you fall in the rapids!)

Sometimes your sequence set may contain many identical or similar sequences — usually a problem because sequences that belong to minority subgroups become harder to align properly. If you can, you want to avoid this situation by removing the sequences yourself, but if you have no choice, it can be reassuring to know that ClustalW is equipped to deal with redundancy.

In case you were wondering, the *W* in *ClustalW* does not stand for Dubya, the U.S. President; it stands for *W*eights. ClustalW uses a sophisticated scheme so that very similar sequences do not end up dominating the multiple sequence alignment. In fact, every sequence is supposed to receive a weight proportional to the amount of new information it contributes.

The most convenient way to use ClustalW is to feed it sequences in FASTA format. However, you can also give to ClustalW a variety of formats, including Swiss-Prot and PIR (the Protein Information Resource format), as well as sequences in the most common multiple-alignment formats.

If you want to use ClustalW effectively, you'll need to observe a couple of caveats:

- **If you give ClustalW a set of sequences already aligned, it does not remove the existing gaps.** This means that the alignment you input influences the alignment ClustalW will produce.
- **The order in which you give sequences to ClustalW sometimes influences the alignment.** If you change this order, even with the same sequences, the alignment may change.

With your sequences in hand, it's now time to fire up the ClustalW server:

1. **Point your browser to the EBI ClustalW server page at `www.ebi.ac.uk/clustalw`.**

   The ClustalW page dutifully appears.

2. **Paste the sequences you collected in the Sequence window.**
3. **Choose Fast from the Alignment pull-down menu (Figure 9-6).**
4. **Use the Output Format pull-down menu to set the selection of your choice.**

   Output formats have various pros and cons. (See Chapter 10 for a discussion on this.) It is safe to use Aln Without Numbers, the default ClustalW format.

   It is never too late to change a format. If you didn't generate your multiple alignment in the format that suits you best, DON'T recompute it! You can easily reformat alignments by using an online reformat utility (such as Fmtseq) at `www.bimcore.emory.edu/Pise/`. (For more on reformatting, see Chapter 10.)

5. **Choose Input from the Output Order pull-down menu. (Refer to Figure 9-6.)**

   Here are a few things to remember about setting output:

WARNING!

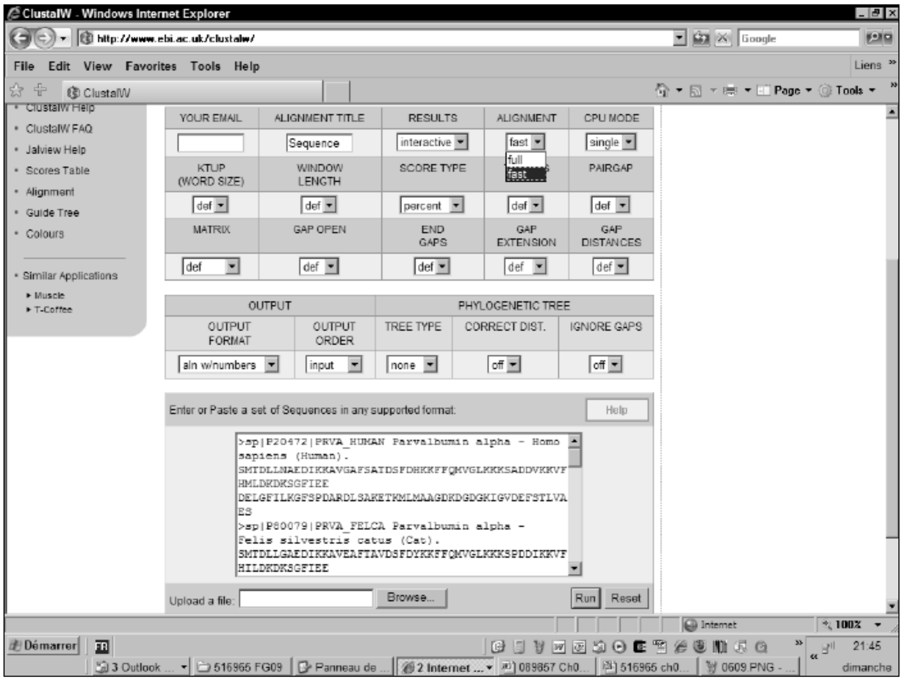
- Although the Aligned output is more informative than Input, Aligned output makes it difficult to compare alignments generated with different methods or different parameters. Usually it's better to prearrange your sequences in the most informative way and *then* select the Input option.

6. **Do not select a tree type.**

   No need to select a tree type here. Any tree computed would be estimated from the unaligned sequences. If you want to compute a tree, compute the alignment first and then turn that alignment into a phylogenetic tree. (See Chapter 13 for more on phylogenetic trees.)

7. **Click the Run button at the bottom of the page.**

   An intermediate page appears. Wait until your browser displays the Results page.



YOUR EMAIL	ALIGNMENT TITLE	RESULTS	ALIGNMENT	CPU MODE
	Sequence	interactive	fast	single
KTUP (WORD SIZE)	WINDOW LENGTH	SCORE TYPE		PAIRGAP
def	def	percent	def	def
MATRIX	GAP OPEN	END GAPS	GAP EXTENSION	GAP DISTANCES
def	def	def	def	def

OUTPUT		PHYLOGENETIC TREE		
OUTPUT FORMAT	OUTPUT ORDER	TREE TYPE	CORRECT DIST.	IGNORE GAPS
aln w/numbers	input	none	off	off

Enter or Paste a set of Sequences in any supported format:

```
>sp|P20472|PRVA_HUMAN Parvalbumin alpha - Homo
sapiens (Human).
SMTDLLNAEDIKKAVGAFSATDSFDHKKFFQMVGLKKKSADDVKKVF
HMLDKDKSGFIEE
DELGFILKGFSPDARDLSAKETKMLMAAGDKDGDGKIGVDEFSTLVA
ES
>sp|P80079|PRVA_FELCA Parvalbumin alpha -
Felis silvestris catus (Cat).
SMTDLLGAEDIKKAVEAFTAVDSFDYKKFFQMVGLKKKSPDDIKKVF
HILDKDKSGFIEE
```

Figure 9-6: The EBI ClustalW server.

- **The multiple alignment:** This section is in the middle of the output. It contains your alignment. You can display this alignment and save it as a text file if you click the hyperlink that comes just before the alignment.

WARNING!

Most ClustalW Web servers (including the EBI) output the ClustalW guide tree (a `.dnd` file). The guide tree is NOT a phylogenetic tree. To obtain a phylogenetic tree from a ClustalW server, you must cut and paste an actual multiple sequence alignment — not a set of unaligned sequences. (We explain how to do this in Chapter 13.)

Changing ClustalW parameters

There are three parameters in ClustalW that can change your alignment: substitution matrices, gap-opening penalties, and gap-extension penalties. (Refer to Figure 9-6 to see the pull-down menus associated with these parameters.) Table 9-5 lists these three types of parameters along with potential consequences when you change them.

Table 9-5 Controlling the Effect of Parameter Tuning in ClustalW

Parameter	*Effect*
Substitution matrix	Substitution matrices control the cost of mutations in sequence alignments. (For more on cost, see Chapter 8.) If you select a category of matrices like PAM or BLOSUM, ClustalW automatically chooses the most adapted index. Predicting the effect of changing matrices is difficult, and there is no such thing as an ideal matrix. If your sequences are closely related, such a change has no effect. If your alignment is difficult to interpret, it may be worth changing from BLOSUM to PAM.
Gap-opening penalty	Gap-opening penalties (GOP) control the cost of opening gaps in your alignment. The higher the value, the more difficult it is to insert a gap in your alignment. The gap-opening penalty is applied once for the opening of each gap. Tuning has little effect because ClustalW readjusts these values automatically.
Gap-extension penalty	Gap-extension penalties control the size of the gaps. It's impossible to predict the optimal couple GOP/GEP, but it's clear that an optimal value exists for almost every protein family — and that the only way to find this value is empirically.

Aligning sequences and structures with Tcoffee

Tcoffee is a recent method developed for conducting multiple sequence alignments. It uses a principle that's a bit similar to ClustalW, but it yields more accurate alignments at the cost of a slightly longer running time. Tcoffee builds a progressive alignment like ClustalW, but it compares segments across the entire sequence set. Table 9-6 lists the main applications of Tcoffee. Aside from its accuracy, the main specificity of Tcoffee is its ability to align sequences and structures (EXPRESSO), the possibility of evaluating the accuracy of an alignment (CORE) and the possibility of combining many alternative multiple sequence alignments into one (Mcoffee).

Table 9-6 Available Tools on www.tcoffee.org

Usage	*Description*
TCOFFEE	Produce a multiple sequence alignment with Tcoffee.
CORE	Evaluate the reliability of an existing multiple alignment.
MCOFFEE	Run any requested Multiple Sequence Alignment package and combine all the output into one final alignment.
EXPRESSO	Incorporate all the available structural information in your alignment. Will produce the best sequence alignments if the structures are available.

Making a multiple alignment with Tcoffee

Making a Multiple sequence alignment with the regular Tcoffee server is only a matter of cutting and pasting your sequences into the right window. (If you're not quite sure how to get the sequences, check out the section "Gathering a known collection of sequences from Swiss-Prot," earlier in this chapter.)

The Build a Multiple Alignment page appears (Figure 9-7).

3. **Paste your sequences into the large window.**

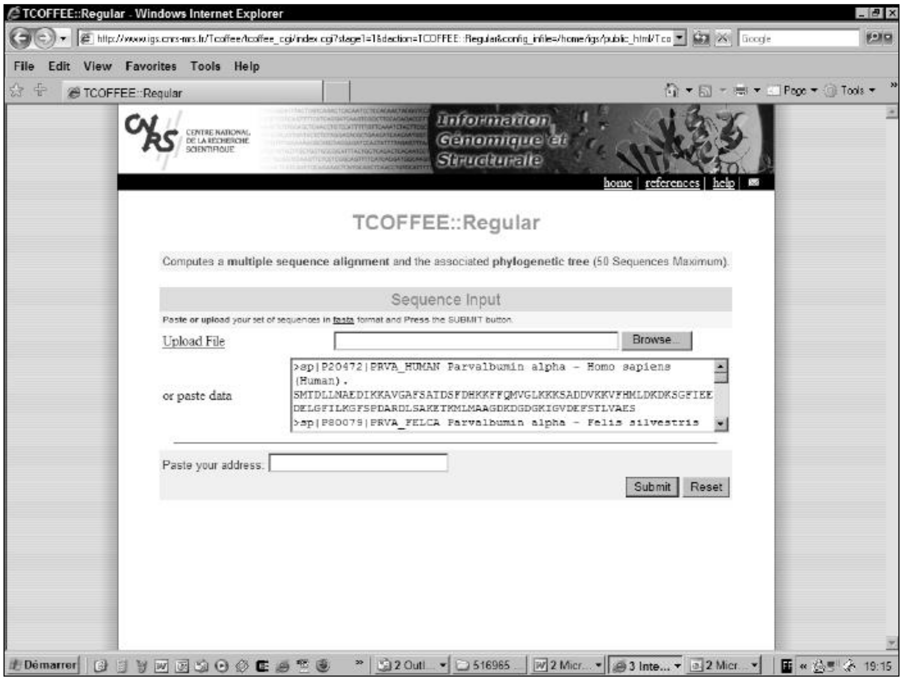
   You can use most formats. If your sequences are in a text file, you can upload this file by using the Browse button.

4. **Click the Submit button at the top or the bottom of the page.**

   Tcoffee can be slow at times. If you'd prefer to be notified when your computation is done, enter your e-mail address in the Web form.

5. **Examine your results.**

   Tcoffee returns a table that contains hyperlinks to your results, as shown in Figure 9-8.


TCOFFEE::Regular - Windows Internet Explorer
http://www.igs.cnrs-mrs.fr/Tcoffee/tcoffee_cgi/index.cgi?stage1=1&daction=TCOFFEE::Regular&config_infile=/home/igs/public_html/Tco
Google
File Edit View Favorites Tools Help
TCOFFEE::Regular
Page
Tools
CNRS CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE
Information Génomique et Structurale
home | references | help
Browse...
Submit
Reset
Démarrer
2 Outl...
516965
2 Micr...
3 Inte...
2 Micr...
19:15

TCOFFEE::Regular

Computes a **multiple sequence alignment** and the associated **phylogenetic tree** (50 Sequences Maximum).

Sequence Input

Paste or upload your set of sequences in fasta format and Press the SUBMIT button.

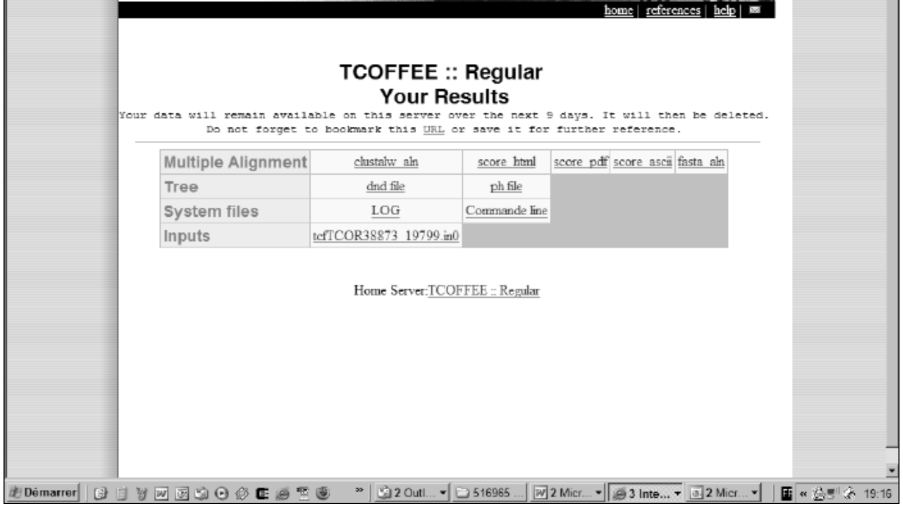
Upload File

or paste data

```
>sp|P20472|PRVA_HUMAN Parvalbumin alpha - Homo sapiens
(Human).
SMTDLLNAEDIKKAVGAFSATDSFDHKKFFQMVGLKKKSADDVKKVFHMLDKDKSGFIEE
DELGFILKGFSPDARDLSAKETKMLMAAGDKDGDGKIGVDEFSTLVAES
>sp|P80079|PRVA_FELCA Parvalbumin alpha - Felis silvestris
```

Paste your address:

Figure 9-7: The Tcoffee Build a Multiple Alignment page.



TCOFFEE :: Regular
Your Results

Your data will remain available on this server over the next 9 days. It will then be deleted.
Do not forget to bookmark this URL or save it for further reference.

Multiple Alignment	clustalw_aln	score_html	score_pdf	score_ascii	fasta_aln
Tree	dnd file	ph file			
System files	LOG	Commande line			
Inputs	tcfTCOR38873_19799.in0				

Home Server:TCOFFEE :: Regular

Figure 9-8: The Tcoffee default output.

The first row of the table is dedicated to multiple sequence alignments and includes

- **msf_aln, clustalw_aln, fasta_aln:** Text files containing your alignment in various formats.

  Keep these files if you want to use your alignment as input for another program.

- **score_html, score_ascii:** A colorized alignment where every residue appears on a background that indicates the quality of this alignment. Red indicates high-quality segments; blue indicates regions of your alignment that you have no reason to trust. The `score_ascii` is a text version of the `.html` file.

  These two last files are meant only for display; you can't use them as an input for other sequence-analysis programs.

The second row is dedicated to phylogenetic trees:

- **dnd:** The guide tree or dendrogram generated by Tcoffee in Newick format (see Chapter 13). You should not use it in place of the true phylogenetic tree.

phylogenetic_tree file.

Combining sequences and structures with EXPRESSO

EXPRESSO is the latest development of Tcoffee, replacing what was known as 3D-Coffee. When you run Expresso, the program uses BLAST to search the PDB (database of structures) for structures whose sequences are similar to your sequences. It then uses theses structures to guide the alignment. Alignments based on structures are expected to be much more accurate than simple sequence alignments.

EXPRESSO is slower than Tcoffee, but if it finds enough structures it produces the most accurate sequence alignments available today, so it's worth a try! EXPRESSO aligns the structures using SAP, a program from Taylor and Orengo, and it aligns sequences and structures using FUGUE, a threading package from Kenji Mizuguchi (developed in Tom Blundell's lab at Cambridge University).

To run EXPRESSO, simply click on the Regular button of the EXPRESSO line on www.tcoffee.org. The rest is identical to Tcoffee — so identical that as a user you'd never know you're using structures!

In the output section, look for a file named template_list. It lists every structure that the program managed to associate with your original sequences. If this file is empty, it means no structure was available to align your sequences and it means your EXPRESSO alignment was merely a standard Tcoffee alignment.

Evaluating the quality of an alignment with CORE

If you want, you can give Tcoffee a multiple alignment that you generated with your favorite method (or by hand if you are a specialist), and you can ask Tcoffee to evaluate the quality of this multiple sequence alignment for you — which can give you an idea of which portions of your alignment you can trust and which are safer to ignore. To do so, cut and paste your alignment into the CORE server on www.tcoffee.org. You can use any of the most common formats (MSF, ALN, FASTA, and PIR).

These evaluations are only empirical, and they do not replace E-values. Nonetheless, it's useful to know that residues with a yellow/orange/red background have an index above 5 (out of 10) — and have more than an 80 percent chance of being correctly aligned.

main difference between Tcoffee and ClustalW is that Tcoffee doesn't directly use substitution matrices to align sequences; it's much lazier, and simply relies on other methods to work for it.

When you give sequences to Tcoffee, it starts making pairwise comparisons. To do so, it takes every possible pair of sequences and makes a global alignment with ClustalW. It also makes a Lalign comparison for each of these pairs. (Lalign is a local-alignment method developed by Huang and Miller to do pairwise local alignments.) Given two sequences, Lalign produces the ten best local alignments. The complete collection of local and global alignments is a *library.*

After the library is finished, Tcoffee builds a multiple alignment that has the highest possible algorithm like ClustalW. This process looks much like a general election, where all the bits of information contained in the library are competing to find their way into the final alignment.

The nice thing about the libraries is that they can contain whatever you want: pairwise alignments, multiple alignments (Mcoffee), global or local, structure-based sequence alignments (EXPRESSO), or even alignments you made yourself by using experimental information. Tcoffee also makes it possible to measure the local agreement between your multiple sequence alignment and any library (CORE).

Crunching large datasets with MUSCLE

MUSCLE is a newcomer in the multiple-sequence-alignment arena — but it is a remarkably efficient package for making fast, high-quality multiple sequence alignments. MUSCLE is ideal if you want to align several hundred sequences. You can access it on various servers, including its home page (at `www.drive5.com/muscle/`). Running MUSCLE is very straightforward — only a matter of cutting and pasting your sequences into the designated window.

Interpreting Your Multiple Sequence Alignment

Interpreting an alignment is a bit of an art. E-values — the scores that tell you how reliable your database search is — do not (yet) exist for multiple sequence alignments. And that means deciding whether your alignment is correct still involves some educated guesswork.

prefer protein alignments.

Recognizing the good parts in a protein alignment

The most convincing evaluative grid we have for a protein multiple alignment stems from our knowledge of protein structures (see Chapter 11). We know that structures contain surface loops that evolve rapidly. Loops are softer portions of the protein that connect its more rigid portions. Protein structures also contain core regions that act as support walls for the protein. These support walls evolve less rapidly than the loops on the surface (see Chapter 11).

In your multiple alignment, you can expect to find nice, gap-free blocks that correspond to the core regions — and gap-rich regions that correspond to the loops. The alignment shown in Figure 9-9 (which appears in the following section, "Taking your multiple alignment further") is a good illustration of this principle.

Now, how can you tell whether a block is good? When you look at a ClustalW, a MUSCLE, or a Tcoffee alignment, you can see that the last line contains seemingly-cabalistic signs such as `(*)`, `(:)`, or `(.)`. These three symbols have very precise meanings:

- `(*)` A star indicates an entirely conserved column.
- `(:)` A colon indicates columns where all the residues have roughly the same size and the same hydropathy.
- `(.)` A period indicates columns where the size *or* the hydropathy has been preserved in the course of evolution.

After you get used to them, these indications are really quite priceless. For instance, your average *good block* is a unit at least 10–30 amino acids long, exhibiting at least one to three stars `(*)`, a few more colons `(:)` close to the stars, and a several periods `(.)` sprinkled here and there. This is exactly what you see in the alignment shown in Figure 9-9.

The magic thing about multiple sequence alignments is that 4 or 5 conserved positions over 50 amino acids can be enough to convince us that we're looking at a genuine signal. This is less than 10 percent identity! If you remember that we require at least 25 percent identity to consider a pairwise alignment amino acids you can expect to see conserved. Amino acids aren't equal — and they all have very characteristic patterns of mutation/conservation in a multiple sequence alignment. Table 9-7 lists the most common features associated with some conserved columns you may come across.

Conserved columns in a multiple sequence alignment are meaningful only when the surrounding columns are not conserved.

Table 9-7 Patterns of Conservation in Multiple Sequence Alignments

Amino Acid	*Characteristic*
W,Y, F	It is common to find conserved tryptophans. Tryptophan is a large hydrophobic residue that sits deep in the core of proteins. It plays an important role in their stability and is therefore difficult to mutate. When tryptophan mutates, it is usually replaced by another aromatic amino acid, such as phenylalanine or tyrosine. Patterns of conserved aromatic amino acids constitute the most common signatures for recognizing protein domains.
G, P	It is common to find conserved columns with a glycine or a proline in a multiple alignment. These two amino acids often coincide with the extremities of well-structured beta strands or alpha helices. (For more on these structures, see Chapter 11.)
C	Cysteines are famous for making C-C (disulphide) bridges. Conserved columns of cysteines are rather common and usually indicate such bridges. Columns of conserved cysteines with a specific distance provide a useful signature for recognizing protein domains and folds.
H, S	Histidine and serine are often involved in catalytic sites, especially those of proteases. Conserved histidine or a conserved serine are good candidates for being part of an active site.
K, R, D, E	These charged amino acids are often involved in ligand binding. Highly conserved columns can also indicate a salt bridge inside the core of the protein.
L	Leucines are rarely very conserved unless they're involved in protein-protein interactions such as a leucine zipper.

when aligning distantly related proteins.

Consider, for instance, the alignment in Figure 9-9. You can use any ClustalW or Tcoffee server to generate this multiple sequence alignment using any server you fancy. It is a good alignment: It contains distantly related proteins, and it is beginning to tell us a nice story about the various components of our protein family. For instance, we can clearly see that the N-terminus region seems to be more conserved than the C-terminus region. In the N-terminus, we can see a short stretch of highly conserved amino acids that make this region a good candidate for being a binding or an active site.

This is interesting — but it isn't enough to make a big story. This alignment still contains too many conserved positions for a detailed analysis. At this point, what we could do is add a few distantly related sequences, one by one, and carefully check the effect of these sequences on the overall alignment quality. More specifically, we want to make sure that these distantly related sequences actually enhance existing patterns rather than completely destroying blocks.

Here is a possible strategy to further reveal the evolutionary constraints within our protein. This strategy relies on the integration within the multiple alignment of precisely those sequences that BLAST reported as marginal hits when we first scanned Swiss-Prot for homologues of the human parvalbumin. (If this all sounds a bit unfamiliar, take a look at the "Selecting sequences on the ExPASy server" section, earlier in this chapter.) You don't actually need to rerun BLAST to use this example; we give you the info you need to know. First and foremost, gather your sequences as follows:

1. **Point your browser to** `www.expasy.ch/sprot/sprot-retrieve-list.html`**.**

   The Swiss-Prot/TrEMBL: Retrieve a List of Entries page appears.

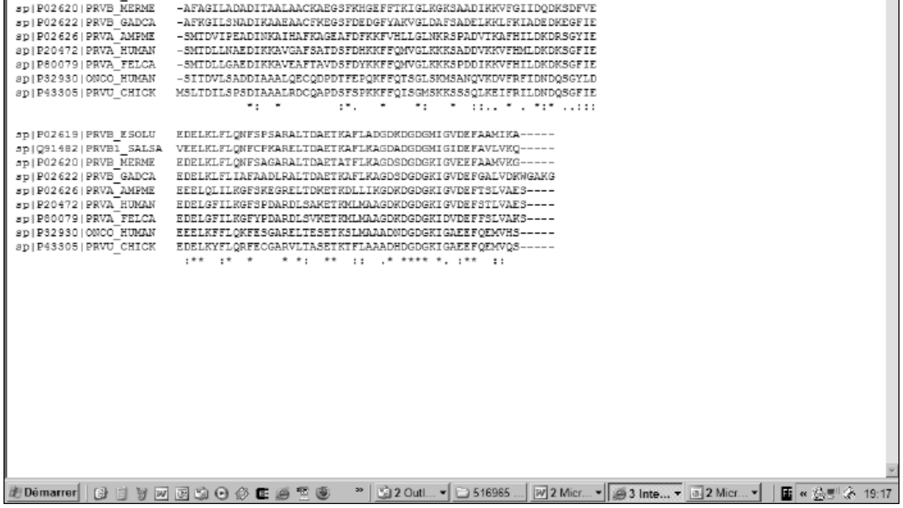
2. **In the Format line, select the FASTA radio button.**

3. **Enter the accession number of your sequences in the Sequence window.**

   Enter one accession number per line. For our example, we entered P20472, P80079, P02626, P02619, P43305, P32930, Q91482, P02620, P02622, P02586.

   P02586 is the TPCS_RABIT, the troponin C of rabbit. In the BLAST of the human parvalbumin against Swiss-Prot that we used to select the other sequences, BLAST reported this hit with an E-value of 5!

   On its own, this result is not interesting — but now that we have a multiple sequence alignment, we can see whether this rabbit can tell us something about our human protein.



```
sp|P02620|PRVB_MERME   -AFAGILADADITAALAACKAEGSFKHGEFFTKIGLKGKSAADIKKVFGIIDQDKSDFVE
sp|P02622|PRVB_GADCA   -AFKGILSNADIKAAEAACFKEGSFDEDGFYAKVGLDAFSADELKKLFKIADEDKEGFIE
sp|P02626|PRVA_AMPME   -SMTDVIPEADINKAIHAFKAGEAFDFKKFVHLLGLNKRSPADVTKAFHILDKDRSGYIE
sp|P20472|PRVA_HUMAN   -SMTDLLNAEDIKKAVGAFSATDSFDHKKFFQMVGLKKKSADDVKKVFHMLDKDKSGFIE
sp|P80079|PRVA_FELCA   -SMTDLLGAEDIKKAVEAFTAVDSFDYKKFFQMVGLKKKSPDDIKKVFHILDKDKSGFIE
sp|P32930|ONCO_HUMAN   -SITDVLSADDIAAALQECQDPDTFEPQKFFQTSGLSKMSANQVKDVFRFIDNDQSGYLD
sp|P43305|PRVU_CHICK   MSLTDILSPSDIAAALRDCQAPDSFSPKKFFQISGMSKKSSSQLKEIFRILDNDQSGFIE
                                   *:  *         :*.    *     *:    *  ::.. * . *:* ..:::

sp|P02619|PRVB_ESOLU   EDELKLFLQNFSPSARALTDAETKAFLADGDKDGDGMIGVDEFAAMIKA-----
sp|Q91482|PRVB1_SALSA  VEELKLFLQNFCPKARELTDAETKAFLKAGDADGDGMIGIDEFAVLVKQ-----
sp|P02620|PRVB_MERME   EDELKLFLQNFSAGARALTDAETATFLKAGDSDGDGKIGVEEFAAMVKG-----
sp|P02622|PRVB_GADCA   EDELKLFLIAFAADLRALTDAETKAFLKAGDSDGDGKIGVDEFGALVDKWGAKG
sp|P02626|PRVA_AMPME   EEELQLILKGFSKEGRELTDKETKDLLIKGDKDGDGKIGVDEFTSLVAES----
sp|P20472|PRVA_HUMAN   EDELGFILKGFSPDARDLSAKETKMLMAAGDKDGDGKIGVDEFSTLVAES----
sp|P80079|PRVA_FELCA   EDELGFILKGFYPDARDLSVKETKMLMAAGDKDGDGKIDVDEFFSLVAKS----
sp|P32930|ONCO_HUMAN   EEELKFFLQKFESGARELTESETKSLMAAADNDGDGKIGAEEFQEMVHS-----
sp|P43305|PRVU_CHICK   EDELKYFLQRFECGARVLTASETKTFLAAADHDGDGKIGAEEFQEMVQS-----
                        :**  :*  *    * *:  **  ::  .* **** *. :**  ::
```

Figure 9-9: A good multiple sequence alignment.

4. **Click the Create FTP File button.**
5. **Copy the sequences onto the Clipboard.**

When you've got all the sequences in your basket (well, on the Clipboard at least), get ready to do some alignment work:

1. **Point your browser to the EBI ClustalW server home page at** `www.ebi.ac.uk/clustalw/index.html`**.**
2. **Paste the sequences you gathered in the preceding steps list into the Sequence window.**
3. **Choose Fast from the Alignment pull-down menu.**
4. **Use the Output Format pull-down menu to choose the output format you want.**
5. **Choose Input from the Output Order pull-down menu.**
6. **Click the Run button at the bottom of the page.**
7. **Save your results.**

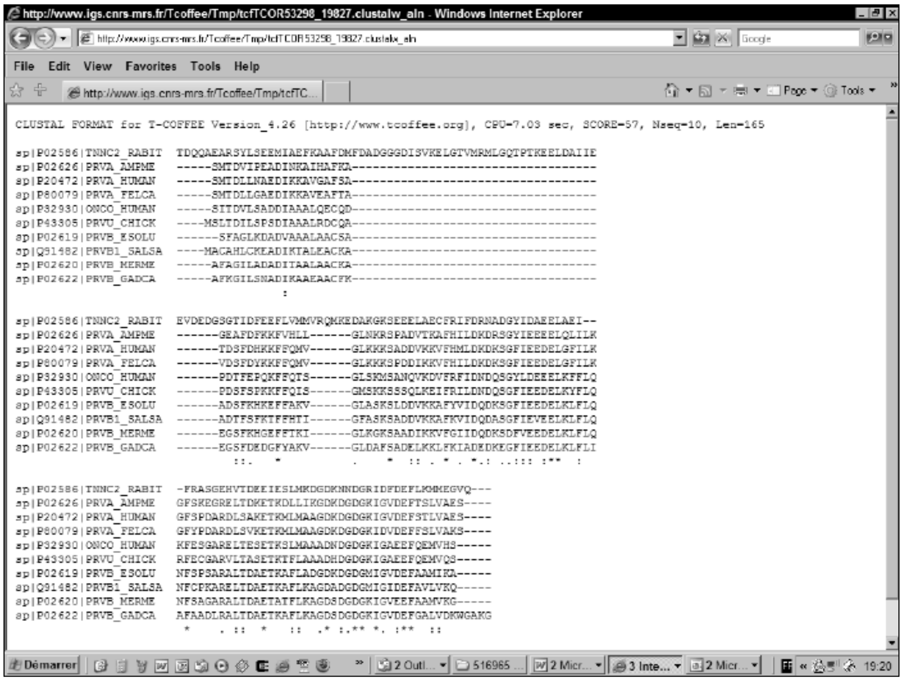
You can see in Figure 9-10 that the new sequence respects the blocks that already existed (Figure 9-9), while shunting some conserved positions. It also reveals regions where insertion and deletions are likely to occur. These the range of possibilities if we were looking for the amino acids responsible for the function of our protein.

At this point, going a bit further and adding another distantly related protein is a good idea. Our aim is to check that these few highly conserved regions are indeed conserved across the whole protein family, even when we compare distantly related cousins.

To do this check, you can go once again through the two previous step lists, using an extra sequence:

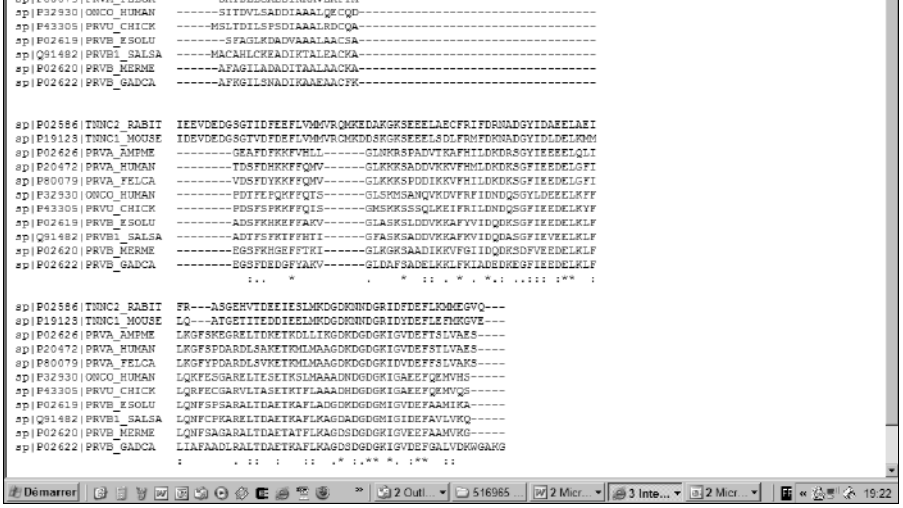
> P20472, P80079, P02626, P02619, P43305, P32930, Q91482, P02620, P02622, P02586, P19123

P19123 is TPCC_MOUSE, the mouse troponin C. It is a very remote homologue of the human parvalbumin. Figure 9-11 shows the result of the inclusion of this new protein in our multiple sequence alignment. It clearly shows that most conserved columns remain.


http://www.igs.cnrs-mrs.fr/Tcoffee/Tmp/tcfTCOR53298_19827.clustalw_aln - Windows Internet Explorer
http://www.igs.cnrs-mrs.fr/Tcoffee/Tmp/tcfTCOR53298_19827.clustalw_aln
Google
File
Edit
View
Favorites
Tools
Help
http://www.igs.cnrs-mrs.fr/Tcoffee/Tmp/tcfTC...
Page
Tools
Démarrer
2 Outl...
516965 ...
2 Micr...
3 Inte...
2 Micr...
19:20

```
CLUSTAL FORMAT for T-COFFEE Version_4.26 [http://www.tcoffee.org], CPU=7.03 sec, SCORE=57, Nseq=10, Len=165

sp|P02586|TNNC2_RABIT  TDQQAEARSYLSEEMIAEFKAAFDMFDADGGGDISVKELGTVMRMLGQTPTKEELDAIIE
sp|P02626|PRVA_AMPME   -----SMTDVIPEADINKAIHAFKA-----------------------------------
sp|P20472|PRVA_HUMAN   -----SMTDLLNAEDIKKAVGAFSA-----------------------------------
sp|P80079|PRVA_FELCA   -----SMTDLLGAEDIKKAVEAFTA-----------------------------------
sp|P32930|ONCO_HUMAN   -----SITDVLSADDIAAALQECQD-----------------------------------
sp|P43305|PRVU_CHICK   ----MSLTDILSPSDIAAALRDCQA-----------------------------------
sp|P02619|PRVB_ESOLU   ------SFAGLKDADVAAALAACSA-----------------------------------
sp|Q91482|PRVB1_SALSA  ----MACAHLCKEADIKTALEACKA-----------------------------------
sp|P02620|PRVB_MERME   -----AFAGILADADITAALAACKA-----------------------------------
sp|P02622|PRVB_GADCA   -----AFKGILSNADIKAAEAACFK-----------------------------------
                                      :

sp|P02586|TNNC2_RABIT  EVDEDGSGTIDFEEFLVMMVRQMKEDAKGKSEEELAECFRIFDRNADGYIDAEELAEI--
sp|P02626|PRVA_AMPME   ------GEAFDFKKFVHLL------GLNKRSPADVTKAFHILDKDRSGYIEEEELQLILK
sp|P20472|PRVA_HUMAN   ------TDSFDHKKFFQMV------GLKKKSADDVKKVFHMLDKDKSGFIEEDELGFILK
sp|P80079|PRVA_FELCA   ------VDSFDYKKFFQMV------GLKKKSPDDIKKVFHILDKDKSGFIEEDELGFILK
sp|P32930|ONCO_HUMAN   ------PDTFEPQKFFQTS------GLSKMSANQVKDVFRFIDNDQSGYLDEEELKFFLQ
sp|P43305|PRVU_CHICK   ------PDSFSPKKFFQIS------GMSKKSSSQLKEIFRILDNDQSGFIEEDELKYFLQ
sp|P02619|PRVB_ESOLU   ------ADSFKHKEFFAKV------GLASKSLDDVKKAFYVIDQDKSGFIEEDELKLFLQ
sp|Q91482|PRVB1_SALSA  ------ADTFSFKTFFHTI------GFASKSADDVKKAFKVIDQDASGFIEVEELKLFLQ
sp|P02620|PRVB_MERME   ------EGSFKHGEFFTKI------GLKGKSAADIKKVFGIIDQDKSDFVEEDELKLFLQ
sp|P02622|PRVB_GADCA   ------EGSFDEDGFYAKV------GLDAFSADELKKLFKIADEDKEGFIEEDELKLFLI
                               ::.   *          .    *  :: . * . *.: ..::: :**  :

sp|P02586|TNNC2_RABIT  -FRASGEHVTDEEIESLMKDGDKNNDGRIDFDEFLKMMEGVQ---
sp|P02626|PRVA_AMPME   GFSKEGRELTDKETKDLLIKGDKDGDGKIGVDEFTSLVAES----
sp|P20472|PRVA_HUMAN   GFSPDARDLSAKETKMLMAAGDKDGDGKIGVDEFSTLVAES----
sp|P80079|PRVA_FELCA   GFYPDARDLSVKETKMLMAAGDKDGDGKIDVDEFFSLVAKS----
sp|P32930|ONCO_HUMAN   KFESGARELTESETKSLMAAADNDGDGKIGAEEFQEMVHS-----
sp|P43305|PRVU_CHICK   RFECGARVLTASETKTFLAAADHDGDGKIGAEEFQEMVQS-----
sp|P02619|PRVB_ESOLU   NFSPSARALTDAETKAFLADGDKDGDGMIGVDEFAAMIKA-----
sp|Q91482|PRVB1_SALSA  NFCPKARELTDAETKAFLKAGDADGDGMIGIDEFAVLVKQ-----
sp|P02620|PRVB_MERME   NFSAGARALTDAETATFLKAGDSDGDGKIGVEEFAAMVKG-----
sp|P02622|PRVB_GADCA   AFAADLRALTDAETKAFLKAGDSDGDGKIGVDEFGALVDKWGAKG
                        *    . ::  *   ::  .* :.** *. :**  ::
```

Figure 9-10: Making a multiple alignment of distantly related proteins.



```
sp|P32930|ONCO_HUMAN   ------SITDVLSADDIAAALQECQD-----------------------------------
sp|P43305|PRVU_CHICK   -----MSLTDILSPSDIAAALRDCQA-----------------------------------
sp|P02619|PRVB_ESOLU   -------SFAGLKDADVAAALAACSA-----------------------------------
sp|Q91482|PRVB1_SALSA  -----MACAHLCKEADIKTALEACKA-----------------------------------
sp|P02620|PRVB_MERME   ------AFAGILADADITAALAACKA-----------------------------------
sp|P02622|PRVB_GADCA   ------AFKGILSNADIKAAEAACFK-----------------------------------


sp|P02586|TNNC2_RABIT  IEEVDEDGSGTIDFEEFLVMMVRQMKEDAKGKSEEELAECFRIFDRNADGYIDAEELAEI
sp|P19123|TNNC1_MOUSE  IDEVDEDGSGTVDFDEFLVMMVRCMKDDSKGKSEEELSDLFRMFDKNADGYIDLDELKMM
sp|P02626|PRVA_AMPME   --------GEAFDFKKFVHLL------GLNKRSPADVTKAFHILDKDRSGYIEEEELQLI
sp|P20472|PRVA_HUMAN   --------TDSFDHKKFFQMV------GLKKKSADDVKKVFHMLDKDKSGFIEEDELGFI
sp|P80079|PRVA_FELCA   --------VDSFDYKKFFQMV------GLKKKSPDDIKKVFHILDKDKSGFIEEDELGFI
sp|P32930|ONCO_HUMAN   --------PDTFEPQKFFQTS------GLSKMSANQVKDVFRFIDNDQSGYLDEEELKFF
sp|P43305|PRVU_CHICK   --------PDSFSPKKFFQIS------GMSKKSSSQLKEIFRILDNDQSGFIEEDELKYF
sp|P02619|PRVB_ESOLU   --------ADSFKHKEFFAKV------GLASKSLDDVKKAFYVIDQDKSGFIEEDELKLF
sp|Q91482|PRVB1_SALSA  --------ADTFSFKTFFHTI------GFASKSADDVKKAFKVIDQDASGFIEVEELKLF
sp|P02620|PRVB_MERME   --------EGSFKHGEFFTKI------GLKGKSAADIKKVFGIIDQDKSDFVEEDELKLF
sp|P02622|PRVB_GADCA   --------EGSFDEDGFYAKV------GLDAFSADELKKLFKIADEDKEGFIEEDELKLF
                                 :..   *          .    *  :: . * . *.: ..::: :**  :

sp|P02586|TNNC2_RABIT  FR---ASGEHVTDEEIESLMKDGDKNNDGRIDFDEFLKMMEGVQ---
sp|P19123|TNNC1_MOUSE  LQ---ATGETITEDDIEELMKDGDKNNDGRIDYDEFLEFMKGVE---
sp|P02626|PRVA_AMPME   LKGFSKEGRELTDKETKDLLIKGDKDGDGKIGVDEFTSLVAES----
sp|P20472|PRVA_HUMAN   LKGFSPDARDLSAKETKMLMAAGDKDGDGKIGVDEFSTLVAES----
sp|P80079|PRVA_FELCA   LKGFYPDARDLSVKETKMLMAAGDKDGDGKIDVDEFFSLVAKS----
sp|P32930|ONCO_HUMAN   LQKFESGARELTESETKSLMAAADNDGDGKIGAEEFQEMVHS-----
sp|P43305|PRVU_CHICK   LQRFECGARVLTASETKTFLAAADHDGDGKIGAEEFQEMVQS-----
sp|P02619|PRVB_ESOLU   LQNFSPSARALTDAETKAFLADGDKDGDGMIGVDEFAAMIKA-----
sp|Q91482|PRVB1_SALSA  LQNFCPKARELTDAETKAFLKAGDADGDGMIGIDEFAVLVKQ-----
sp|P02620|PRVB_MERME   LQNFSAGARALTDAETATFLKAGDSDGDGKIGVEEFAAMVKG-----
sp|P02622|PRVB_GADCA   LIAFAADLRALTDAETKAFLKAGDSDGDGKIGVDEFGALVDKWGAKG
                       :       . ::  :   ::  .* :.** *. :**  ::
```

Figure 9-11: Including more distant relatives in the multiple alignment.

With such a result, we can safely bet that these two conserved regions are surely involved in some biological function. We know that most of these proteins bind calcium, so we can safely bet that the calcium-binding site involves some of these conserved positions. This is indeed the case as revealed by the Swiss-Prot annotation.

Comparing Sequences That You Can't Align

Sometimes you need to compare sequences that don't necessarily have a common ancestor — or are so distantly related that considering them as homologous is difficult.

Multiple-sequence-alignment programs are usually hopeless in these situations. They'll include proteins that contain homologous domains but that are otherwise unrelated, or segments of DNA that seem to contain similar regulating features although it is impossible to see anything common between them in a multiple alignment.

method, looks for flexible patterns — a special category of segments that can contain gaps, and that need to be conserved only at certain positions.

Making multiple local alignments with the Gibbs sampler

The Gibbs sampler is a stochastic method: It first scrambles your sequences, aligns them randomly until a good solution appears, and then keeps scrambling the sequences to improve this good solution. The interesting thing about using chance is that you can solve complicated problems without having to explore all the possible solutions.

Gurus name this type of method *stochastic* because it contains an element of chance. Stochastic methods are by far the most powerful in bioinformatics. Unfortunately, they do not have the popularity they deserve because they make it difficult to reproduce the same results twice. For instance, if you run the Gibbs sampler twice on the same set of sequences, you may not get exactly the same solution. For most biologists, reproducibility is paramount, and — let's face it — we all hate it when computers change their minds from one minute to the next.

Nonetheless, if you're ready for a bit of fuzzy logic, you may find that the Gibbs sampler can offer very sensible solutions to extremely complicated problems. For instance, the Gibbs sampler is very good at identifying HTH (Helix-Turn-Helix) domains across a protein family. It is also a nice way to search for regulatory elements shared by otherwise unrelated DNA sequences. You can access the Gibbs sampler from its home page at

```
bayesweb.wadsworth.org/gibbs/gibbs.html
```

or use the server the Pasteur Institute maintains at

```
bioweb.pasteur.fr/seqanal/interfaces/gibbs-simple.html
```

When using a Gibbs sampler, bear in mind that to be accurate, it needs as many sequences as possible. You should not use it with less than 20 sequences.

doing bioinformatics have different lengths; the most standard situation is to find motifs that are poorly conserved. When this happens, your only hope is to find a few highly conserved amino acids or nucleotides that anchor the motif.

If your sequences are poorly related but contain such motifs, then the only way to analyze them is to use a pattern-finding motif. If you want to know more about patterns (biologists often call them PROSITE patterns), you can go to Chapter 7. Several tools exist for identifying these conserved patterns in a set of unaligned sequences. Showing you how to use these tools and to interpret their result is beyond the scope of this book. However, if you want to experiment with them, you should know that Pratt is one of the most powerful because it allows some flexible spacing between the conserved positions. You can also use TEIRESIAS, MEME, or SMILE. (Flip ahead to Table 9-10 to see a partial list of these resources.)

Internet Resources for Doing Multiple Sequence Comparisons

The amount of resources for making multiple alignments online is almost overwhelming. The usual words of caution apply to the resources that we list in the following tables:

- Use stable resources and always make a few simple tests to make sure that the service you're using does what it's supposed to do.
- Never depend blindly on a resource you don't control. As they say, "Good servers go bad, and bad servers go down."
- If you want to do many multiple sequence alignments — or if your company does not authorize you to send your sequences over the Internet — you may have to install these programs on your own machine. We give you addresses where you can download the source code or the executable files of some of these programs.

available servers, starting with the site where you can download ClustalW to install it on your own machine.

Note: The last link in Table 9-8 is the location where you can download the executable to install it on your own machine. It is *not* a Web server.

Table 9-8 A List of ClustalW Servers

Name	*Location*	*Address*
EBI	Europe	`www.ebi.ac.uk/clustalw`
EMBnet	Europe	`www.ch.embnet.org/software/ClustalW.html`
PIR	USA	`pir.georgetown.edu/pirwww/search/multialn.shtml`
BCM	USA	`searchlauncher.bcm.tmc.edu/multi-align/multi-align.html`
GenomeNet	Japan	`align.genome.jp/`
DDBJ	Japan	`www.ddbj.nig.ac.jp/search/clustalw-e.html`
Strasbourg	Europe	`ftp://ftp-igbmc.u-strasbg.fr/pub/ClustalW/`

Finding your favorite alignment method

If ClustalW isn't suited for the sequences you want to align, Table 9-9 saves the day by giving you the addresses of a few other servers.

	of sequences and structures	www.ch.embnet.org/ software/TCoffee.html www.ebi.ac.uk/t-coffee/
Probcons	A Bayesian version of Tcoffee	probcons.stanford.edu/
MUSCLE	A fast and accurate sequence cruncher	www.drive5.com/muscle/
Kalign	A fast sequence aligner	msa.cgb.ki.se
MAFFT	A fast and accurate sequence cruncher using Fast Fourier Transforms	timpani.genome.ad.jp/~ mafft/server/
Dialign	Ideal for Sequences With Local Homology	bibiserv.techfak. uni-bielefeld.de/dialign/

Searching for motifs or patterns

If your sequences are too distantly related for making a multiple sequence alignment, your last chance is trying to find a conserved pattern across them. Table 9-10 lists some of the online methods you can use for this purpose.

Table 9-10 Motif-finding Methods Available Online

Method	*Address*
Gibbs Sampler	bioweb.pasteur.fr/seqanal/interfaces/ gibbs-simple.html, bayesweb.wadsworth. org/gibbs/gibbs.html
Pratt	http://www.ebi.ac.uk/pratt/
eMotif	dna.stanford.edu/emotif/

(continued)

TEIRESIAS	http://cbcsrv.watson.ibm.com/Tspd.html
Bioprospector	http://ai.stanford.edu/~xsliu/BioProspector/
Improbizer	www.soe.ucsc.edu/~kent/improbizer/improbizer.html
BLOCK-Maker	blocks.fhcrc.org/blocks/blockmkr/make_blocks.html

Editing and Publishing Alignments

In This Chapter

- Reformatting your multiple sequence alignment
- Manually editing a multiple sequence alignment with Jalview
- Turning your Alignment into a logo
- Shading a multiple sequence alignment with Boxshade

It looked so good that I thought it just had to be genuine.

— Everyone's favorite secret thought

This chapter is all about using and modifying an existing multiple sequence alignment. If you don't have one ready yet, don't worry; we start by showing you how to generate a dummy multiple sequence alignment that you can use throughout the entire chapter.

The first real section in this chapter — as opposed to the kind-of-real section about making a dummy multiple sequence alignment — tells you what gurus assume we all know about alignment formats. We also show you how to decide which format you want to use (or avoid!) and — generally speaking — how to find your way across the format jungle.

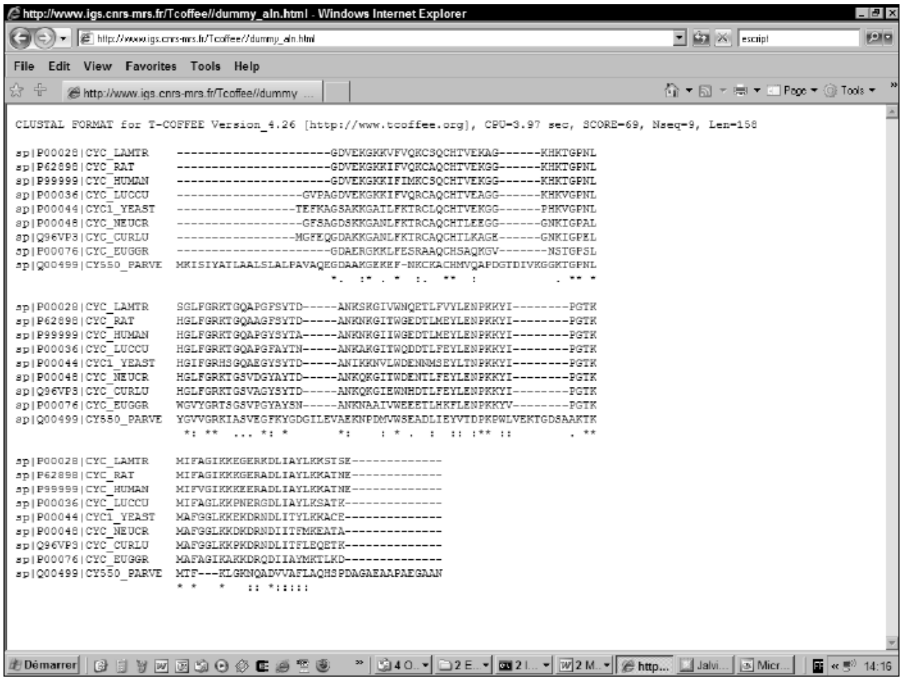
If you have generated your multiple sequence alignment with any kind of automatic method, you probably need to arrange (edit) it manually before you can really use it. When you stop to think about it, editing a multiple sequence alignment isn't trivial: It requires inserting gaps in an entire subgroup of sequences or shifting several sequences all at once, and so on. When you do this with a standard word processor, pretty soon you're going to feel as if you're playing with a Rubik's Cube. No matter how slowly and carefully you start, after three minutes of moving things about, everything is out of control! In order to avoid this kind of confusion, we show you some powerful editing tools in this chapter that you can use over the Internet.

them in this chapter.

Finally, we show you a couple of tools that can help you extract information from your multiple sequence alignment. These include using protein logos to identify the most conserved positions.

All in all, this is a small chapter — but if you decide to go through it, you'll have no more excuses for attaching simple, raw-text alignments to your reports, publications, or lab books!

If you do not have an alignment ready, you can download one for your crash tests from `www.tcoffee.org/dummy_aln.html`. It is an alignment in ALN format (see Figure 10-1).


http://www.igs.cnrs-mrs.fr/Tcoffee//dummy_aln.html - Windows Internet Explorer
http://www.igs.cnrs-mrs.fr/Tcoffee//dummy_aln.html
escript
File
Edit
View
Favorites
Tools
Help
http://www.igs.cnrs-mrs.fr/Tcoffee//dummy ...
Page
Tools
Démarrer
4 O...
2 E...
2 I...
2 M...
http...
Jalvi...
Micr...
14:16

```
CLUSTAL FORMAT for T-COFFEE Version_4.26 [http://www.tcoffee.org], CPU=3.97 sec, SCORE=69, Nseq=9, Len=158

sp|P00028|CYC_LAMTR    ----------------------GDVEKGKKVFVQKCSQCHTVEKAG------KHKTGPNL
sp|P62898|CYC_RAT      ----------------------GDVEKGKKIFVQKCAQCHTVEKGG------KHKTGPNL
sp|P99999|CYC_HUMAN    ----------------------GDVEKGKKIFIMKCSQCHTVEKGG------KHKTGPNL
sp|P00036|CYC_LUCCU    ------------------GVPAGDVEKGKKIFVQRCAQCHTVEAGG------KHKVGPNL
sp|P00044|CYC1_YEAST   -----------------TEFKAGSAKKGATLFKTRCLQCHTVEKGG------PHKVGPNL
sp|P00048|CYC_NEUCR    ------------------GFSAGDSKKGANLFKTRCAQCHTLEEGG------GNKIGPAL
sp|Q96VP3|CYC_CURLU    -----------------MGFEQGDAKKGANLFKTRCAQCHTLKAGE------GNKIGPEL
sp|P00076|CYC_EUGGR    ----------------------GDAERGKKLFESRAAQCHSAQKGV-------NSTGPSL
sp|Q00499|CY550_PARVE  MKISIYATLAALSLALPAVAQEGDAAKGEKEF-NKCKACHMVQAPDGTDIVKGGKTGPNL
                                             *.  :* . *  :.  **  :           . ** *

sp|P00028|CYC_LAMTR    SGLFGRKTGQAPGFSYTD-----ANKSKGIVWNQETLFVYLENPKKYI--------PGTK
sp|P62898|CYC_RAT      HGLFGRKTGQAAGFSYTD-----ANKNKGITWGEDTLMEYLENPKKYI--------PGTK
sp|P99999|CYC_HUMAN    HGLFGRKTGQAPGYSYTA-----ANKNKGIIWGEDTLMEYLENPKKYI--------PGTK
sp|P00036|CYC_LUCCU    HGLFGRKTGQAPGFAYTN-----ANKAKGITWQDDTLFEYLENPKKYI--------PGTK
sp|P00044|CYC1_YEAST   HGIFGRHSGQAEGYSYTD-----ANIKKNVLWDENNMSEYLTNPKKYI--------PGTK
sp|P00048|CYC_NEUCR    HGLFGRKTGSVDGYAYTD-----ANKQKGITWDENTLFEYLENPKKYI--------PGTK
sp|Q96VP3|CYC_CURLU    HGLFGRKTGSVAGYSYTD-----ANKQKGIEWNHDTLFEYLENPKKYI--------PGTK
sp|P00076|CYC_EUGGR    WGVYGRTSGSVPGYAYSN-----ANKNAAIVWEEETLHKFLENPKKYV--------PGTK
sp|Q00499|CY550_PARVE  YGVVGRKIASVEGFKYGDGILEVAEKNPDMVWSEADLIEYVTDPKPWLVEKTGDSAAKTK
                        *: **  ... *: *       *:    : * .  :  :: :** ::        . **

sp|P00028|CYC_LAMTR    MIFAGIKKEGERKDLIAYLKKSTSE-------------
sp|P62898|CYC_RAT      MIFAGIKKKGERADLIAYLKKATNE-------------
sp|P99999|CYC_HUMAN    MIFVGIKKKEERADLIAYLKKATNE-------------
sp|P00036|CYC_LUCCU    MIFAGLKKPNERGDLIAYLKSATK--------------
sp|P00044|CYC1_YEAST   MAFGGLKKEKDRNDLITYLKKACE--------------
sp|P00048|CYC_NEUCR    MAFGGLKKDKDRNDIITFMKEATA--------------
sp|Q96VP3|CYC_CURLU    MAFGGLKKPKDRNDLITFLEQETK--------------
sp|P00076|CYC_EUGGR    MAFAGIKAKKDRQDIIAYMKTLKD--------------
sp|Q00499|CY550_PARVE  MTF---KLGKNQADVVAFLAQHSPDAGAEAAPAEGAAN
                       * *   *   :: *:::::
```

Figure 10-1: A multiple sequence alignment in ALN or ClustalW format.

When using online servers — or even local programs — you do not always have full control of what goes in and what comes out of the program you use. For instance, many multiple-sequence-alignment programs commonly output only one format: MSF (Multiple Sequence Format). What can you do if you want to analyze this multiple sequence alignment with a program that only reads FASTA-formatted alignments? The answer is simple: Use a reformatting program just like the ones we introduce in this chapter.

When it comes to multiple sequence alignments, there is no such thing as a perfect format. Anyone used to dealing with the many services available online knows that each service tends to have its favorite format. (See the "A format for everyone: Democracy or anarchy?" sidebar if you want proof of this fact.) Understanding everything there is to know about formats is an ability you only acquire with some practice — but this chapter gives you some ideas, and we prepare you to be on the lookout for potential problems.

A format for everyone: Democracy or anarchy?

The variety of formats is a major curse in bioinformatics. For a long time, the tradition was that anyone developing a new program would design his or her own house format. Each new format was slightly different from the others — and especially appealing to only a particular category of biologists. For instance, people doing phylogeny became very attached to the Phylip format; users of GCG (a popular bioinformatics package) preferred MSF, and so on. Whether we like it or not, the weight of history has made each of these formats totally acceptable — and today they all live in perfect harmony, side by side on our computer keyboards — or do they?

Another source of confusion in the field was the communication wars between biologists and computer scientists. For many years, biologists have complained about the computer scientists' formats, basically saying, "We cannot make sense of this gibberish." In retaliation (and because they needed to), biologists also created formats they could use — to the contempt of computer scientists, who dubbed these formats "amateurish and ambiguous." Both the biologists and the computer scientists were probably right in their respective evaluations. But that didn't help.

Things may be about to change with the XML language coming to the fore. XML (e*X*tensible *M*arkup *L*anguage) is a close relative of HTML — the language of the Web. XML makes it possible to simply describe your data with keywords everyone can agree on. Today, biologists and computer scientists consent — albeit weakly — that XML could be the solution and the main bioinformatics programs are now able to produce output in XML.

yourself four questions:

- Do most programs support this format?
- Will my collaborators be able to use it?
- Can I store all the information I need with this format?
- Is it easy to manipulate?

Today no format is so perfect that you would answer yes to each of these questions; any choice relies on the right trade-off between what you really need and what's available. If you can, avoid using a mixture of formats when running a project.

The taxonomy of multiple-sequence-alignment formats is rather complex. Table 10-1 lists the most common formats you may come across when manipulating multiple sequence alignments.

Table 10-1 A Classification of Multiple Sequence Alignment Formats

Name	*Type*	*Usage*
post-script, PDF, HTML	Graphic	Terminal formats suitable for printing only
FASTA (Figure 10-2)	Text	Easy to manipulate Noninterleaved (not readable by humans) Easy for renaming sequences Supported by most programs Cannot incorporate extra annotation
PIR	Text	Similar to FASTA but with an extra line Can incorporate limited extra annotation
MSF (Figure 10-3)	Text	Most standard multiple-alignment format Difficult to modify manually Interleaved (easy to read for humans) Can include extra annotation, such as weights Supported by most programs (but some programs only support it partially!)

ALN (Figure 10-1)	Text	Simplified version of MSF Default output of ClustalW Interleaved (easy to read for humans) Supported by many programs
Phylip	Text	Variant of ALN Useful for doing phylogenetic analysis Supported by most phylogenetic packages

Recognizing the main formats

The alignments you see in publications are usually in a graphic format that isn't very useful for doing analyses. The last section of this chapter shows you resources that you can use for generating these alignments. You've already seen the ALN format in this chapter (refer to Figure 10-1), and FASTA is another format (see Figure 10-2) that has cropped up a number of times in previous chapters. In the lab, MSF, as shown in Figure 10-3, is probably the most common format. Note that MSF and ALN are *interleaved* (meaning easier for humans to read but not for machines), whereas FASTA is not.

REMEMBER

To make an interleaved format, the idea is to take the complete multiple sequence alignment — and then chop it into blocks of 60 residues. The file displays the blocks one after the other.

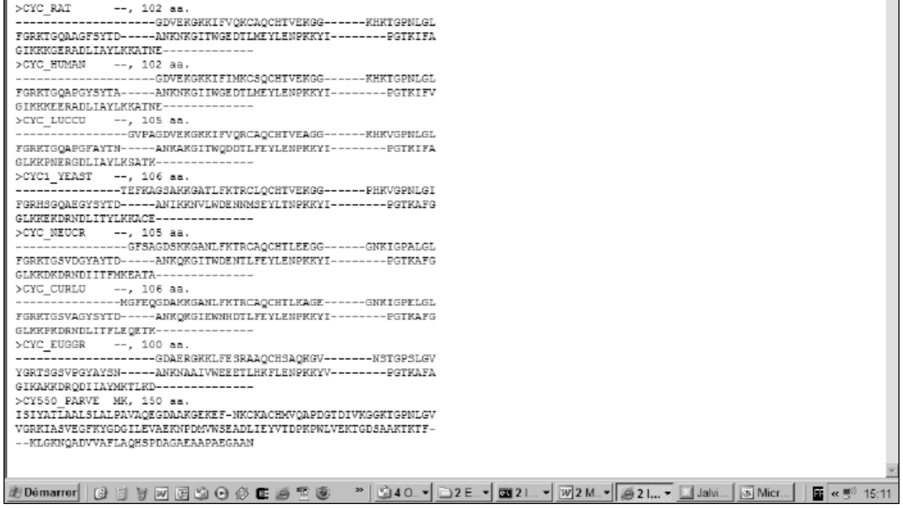
TIP

If you want to further investigate the intricacies of the world of formatting, here are two extensive resources for keeping your formats straight: `emboss.sourceforge.net/docs/themes/SequenceFormats.html` and `www.ebi.ac.uk/help/formats_frame.html`.

Working with the right format

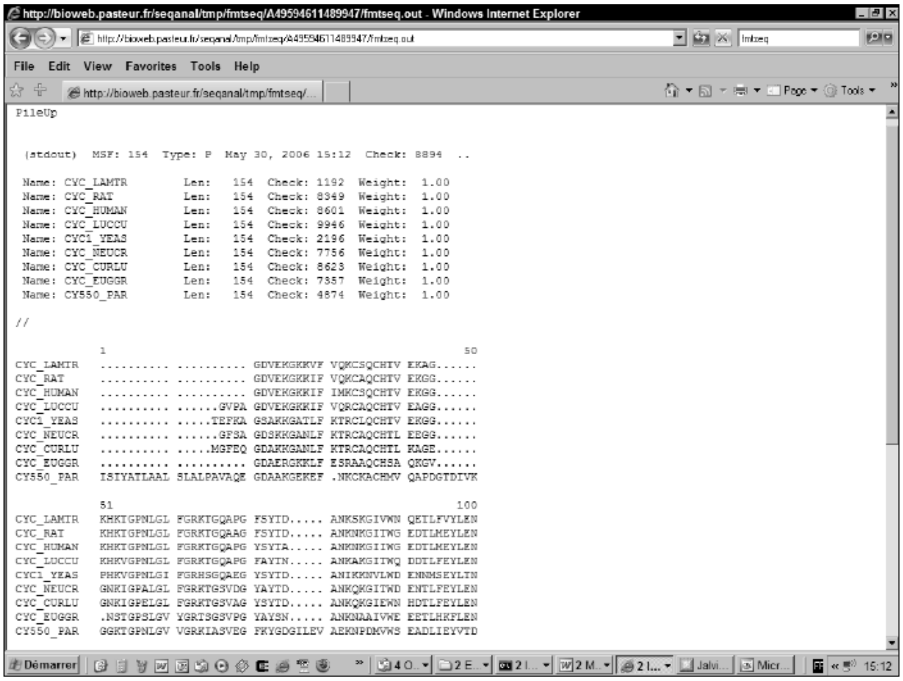
WARNING!

Sometimes when a program offers several outputs, keeping only the graphic output is tempting because it seems — with its flashy colored annotation — to be the most complete. This graphic output usually comes in post-script, `.pdf`, `.html`, `.gif`, or `.jpeg` format. If you keep only this graphic output, though, you can be badly stuck.



```
>CYC_RAT      --, 102 aa.
--------------------GDVEKGKKIFVQKCAQCHTVEKGG------KHKTGPNLGL
FGRKTGQAAGFSYTD-----ANKNKGITWGEDTLMEYLENPKKYI--------PGTKIFA
GIKKKGERADLIAYLKKATNE-------------
>CYC_HUMAN    --, 102 aa.
--------------------GDVEKGKKIFIMKCSQCHTVEKGG------KHKTGPNLGL
FGRKTGQAPGYSYTA-----ANKNKGIIWGEDTLMEYLENPKKYI--------PGTKIFV
GIKKKEERADLIAYLKKATNE-------------
>CYC_LUCCU    --, 105 aa.
----------------GVPAGDVEKGKKIFVQRCAQCHTVEAGG------KHKVGPNLGL
FGRKTGQAPGFAYTN-----ANKAKGITWQDDTLFEYLENPKKYI--------PGTKIFA
GLKKPNERGDLIAYLKSATK--------------
>CYC1_YEAST   --, 106 aa.
---------------TEFKAGSAKKGATLFKTRCLQCHTVEKGG------PHKVGPNLGI
FGRHSGQAEGYSYTD-----ANIKKNVLWDENNMSEYLTNPKKYI--------PGTKAFG
GLKKEKDRNDLITYLKKACE--------------
>CYC_NEUCR    --, 105 aa.
----------------GFSAGDSKKGANLFKTRCAQCHTLEEGG------GNKIGPALGL
FGRKTGSVDGYAYTD-----ANKQKGITWDENTLFEYLENPKKYI--------PGTKAFG
GLKKDKDRNDIITFMKEATA--------------
>CYC_CURLU    --, 106 aa.
---------------MGFEQGDAKKGANLFKTRCAQCHTLKAGE------GNKIGPELGL
FGRKTGSVAGYSYTD-----ANKQKGIEWNHDTLFEYLENPKKYI--------PGTKAFG
GLKKPKDRNDLITFLEQETK--------------
>CYC_EUGGR    --, 100 aa.
--------------------GDAERGKKLFESRAAQCHSAQKGV-------NSTGPSLGV
YGRTSGSVPGYAYSN-----ANKNAAIVWEEETLHKFLENPKKYV--------PGTKAFA
GIKAKKDRQDIIAYMKTLKD--------------
>CY550_PARVE  MK, 150 aa.
ISIYATLAALSLALPAVAQEGDAAKGEKEF-NKCKACHMVQAPDGTDIVKGGKTGPNLGV
VGRKIASVEGFKYGDGILEVAEKNPDMVWSEADLIEYVTDPKPWLVEKTGDSAAKTKTF-
--KLGKNQADVVAFLAQHSPDAGAEAAPAEGAAN
```

Figure 10-2: Multiple sequence alignment in FASTA format.



http://bioweb.pasteur.fr/seqanal/tmp/fmtseq/A49594611489947/fmtseq.out - Windows Internet Explorer
http://bioweb.pasteur.fr/seqanal/tmp/fmtseq/A49594611489947/fmtseq.out
fmtseq
File
Edit
View
Favorites
Tools
Help
http://bioweb.pasteur.fr/seqanal/tmp/fmtseq/...
Page
Tools
Démarrer
15:12

```
PileUp


 (stdout)  MSF: 154  Type: P  May 30, 2006 15:12  Check: 8894  ..

 Name: CYC_LAMTR        Len:   154  Check: 1192  Weight:  1.00
 Name: CYC_RAT          Len:   154  Check: 8349  Weight:  1.00
 Name: CYC_HUMAN        Len:   154  Check: 8601  Weight:  1.00
 Name: CYC_LUCCU        Len:   154  Check: 9946  Weight:  1.00
 Name: CYC1_YEAS        Len:   154  Check: 2196  Weight:  1.00
 Name: CYC_NEUCR        Len:   154  Check: 7756  Weight:  1.00
 Name: CYC_CURLU        Len:   154  Check: 8623  Weight:  1.00
 Name: CYC_EUGGR        Len:   154  Check: 7357  Weight:  1.00
 Name: CY550_PAR        Len:   154  Check: 4874  Weight:  1.00

//

            1                                                   50
CYC_LAMTR   .......... .......... GDVEKGKKVF VQKCSQCHTV EKAG......
CYC_RAT     .......... .......... GDVEKGKKIF VQKCAQCHTV EKGG......
CYC_HUMAN   .......... .......... GDVEKGKKIF IMKCSQCHTV EKGG......
CYC_LUCCU   .......... ......GVPA GDVEKGKKIF VQRCAQCHTV EAGG......
CYC1_YEAS   .......... .....TEFKA GSAKKGATLF KTRCLQCHTV EKGG......
CYC_NEUCR   .......... ......GFSA GDSKKGANLF KTRCAQCHTL EEGG......
CYC_CURLU   .......... .....MGFEQ GDAKKGANLF KTRCAQCHTL KAGE......
CYC_EUGGR   .......... .......... GDAERGKKLF ESRAAQCHSA QKGV......
CY550_PAR   ISIYATLAAL SLALPAVAQE GDAAKGEKEF .NKCKACHMV QAPDGTDIVK

            51                                                 100
CYC_LAMTR   KHKTGPNLGL FGRKTGQAPG FSYTD..... ANKSKGIVWN QETLFVYLEN
CYC_RAT     KHKTGPNLGL FGRKTGQAAG FSYTD..... ANKNKGITWG EDTLMEYLEN
CYC_HUMAN   KHKTGPNLGL FGRKTGQAPG YSYTA..... ANKNKGIIWG EDTLMEYLEN
CYC_LUCCU   KHKVGPNLGL FGRKTGQAPG FAYTN..... ANKAKGITWQ DDTLFEYLEN
CYC1_YEAS   PHKVGPNLGI FGRHSGQAEG YSYTD..... ANIKKNVLWD ENNMSEYLTN
CYC_NEUCR   GNKIGPALGL FGRKTGSVDG YAYTD..... ANKQKGITWD ENTLFEYLEN
CYC_CURLU   GNKIGPELGL FGRKTGSVAG YSYTD..... ANKQKGIEWN HDTLFEYLEN
CYC_EUGGR   .NSTGPSLGV YGRTSGSVPG YAYSN..... ANKNAAIVWE EETLHKFLEN
CY550_PAR   GGKTGPNLGV VGRKIASVEG FKYGDGILEV AEKNPDMVWS EADLIEYVTD
```

Figure 10-3: A multiple sequence alignment in MSF format.

Text representations are much more civilized; they are the default output of most multiple-sequence-alignment programs and servers. Their main purpose is to be easy to display, easy to manipulate, and easy to transfer from one program to the next. Of course, they're much less flashy than their graphic counterparts!

MSF is the most standard format for multiple sequence alignments. This is the most common format if you download an alignment from the Web, for example. FASTA or PIR are a bit less common but are on the rise. If you're using phylogenetic programs, Phylip is fairly common. ALN is also fairly common because it is the default output of ClustalW, the most widely used multiple-sequence-alignment program. ALN offers a good trade-off between most formats.

Converting formats

If the program you're using doesn't produce alignments in the format you need, it is possible to use a third-party conversion tool to get to the format you want. In the following step list, we show you how to convert an alignment in ALN format (refer to Figure 10-1) into a FASTA-format multiple alignment.

1. **Point your browser to** `bioweb.pasteur.fr/seqanal/interfaces/fmtseq.html`.

   The Pasteur Institute's fmtseq Sequence Conversion page appears, as shown in Figure 10-4.

2. **Enter your e-mail address in the field provided.**

3. **Paste your alignment in the Sequence window.**

   fmtseq automatically recognizes most formats. If you don't have any spare sequences lying around, use the dummy multiple sequence alignment we created at the beginning of the chapter.

   If your sequence names are longer than 15 characters, the rest of the name gets incorporated IN the sequence. This is a major source of confusion.

4. **Specify a format for your output by making a selection from the Output Sequence Format drop-down menu.**

   For our example, we chose FASTA.

5. **Scroll Down The Page to reach the "Input Parameters" section (you can also click on the Input Parameters link) and select the format you want to input from the Input Sequence Format drop-down menu.**

When you deselect these two options, fmtseq retains the original case of each symbol in your alignment. Some programs, such as Dialign, use case to indicate the local reliability of the alignment.

TIP

The Pretty-Print option gives you access to a menu that makes it possible to control every detail in your output format, such as the number of characters per line, the indentation, and other minute details.

7. **Click the Run ftmseq button at the top or the bottom of the page.**

   Wait until your browser displays the result page. In most cases, the results come back interactively in a few seconds, but if the server is very busy, it may send the output using the e-mail you provided in Step 2.

8. **Click the** <u>fmtseq.out</u> **link to see your alignment reformatted.**

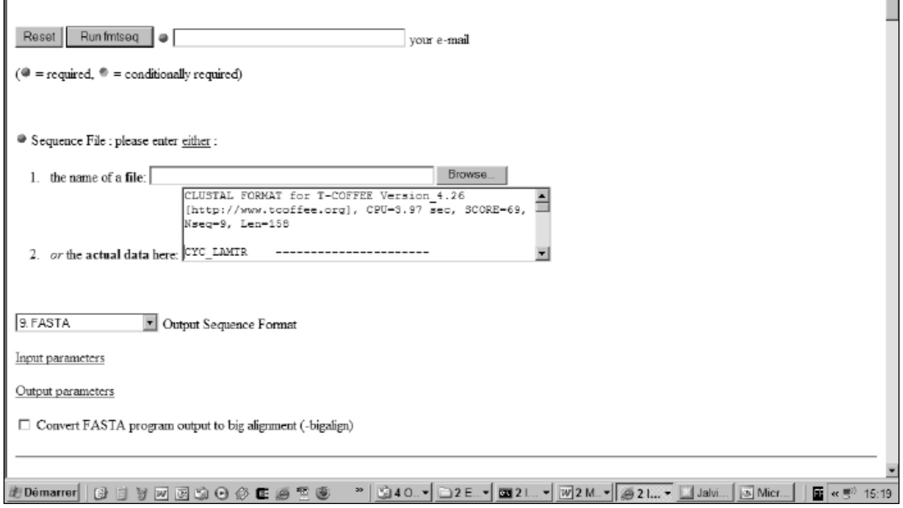
9. **Save the reformatted file with the File➪Save As option of your browser.**

   You can save this file as a `.txt` or `.html` file.

Sequence reformatting is one of the most common facilities on the Web. Table 10-2 lists some of these sequence reformatting servers and what you can do using them.

Table 10-2 Sequence Text Conversion on the Web

Name	*Address*	*Description*
fmtseq	`bioweb.pasteur.fr/seqanal/interfaces/fmtseq-simple.html`, `www.bimcore.emory.edu/Pise/`	Converts most formats
READSEQ	`bimas.dcrt.nih.gov/molbio/readseq/`, `dot.imgen.bcm.tmc.edu/seq-util/readseq.html`, `iubio.bio.indiana.edu/cgi-bin/readseq.cgi`, `bimas.dcrt.nih.gov/molbio/readseq/`, `www.ebi.ac.uk/readseq/`	A very popular tool for reformatting; does not recognize ALN
SeqCheck	`darwin.nmsu.edu/bioinfo/seqcheck/seqcheck.php`	Cleans your FASTA sequences



● your e-mail

(● = required, ● = conditionally required)

● Sequence File : please enter either :

1. the name of a file:
2. or the actual data here:

```
CLUSTAL FORMAT for T-COFFEE Version_4.26
[http://www.tcoffee.org], CPU=3.97 sec, SCORE=69,
Nseq=9, Len=158

CYC_LAMTR    ----------------------
```

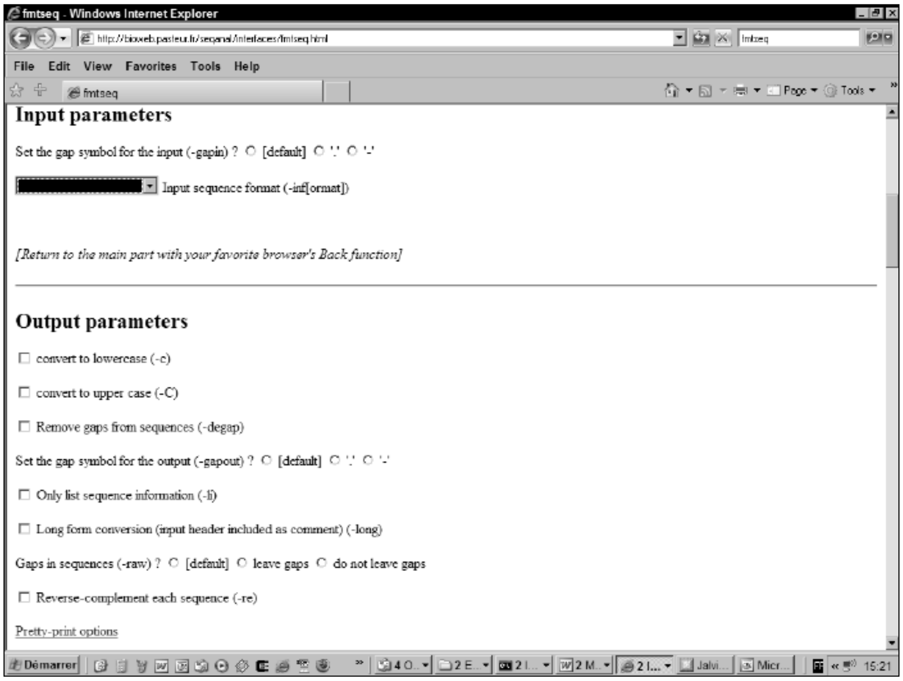
9. FASTA Output Sequence Format

Input parameters

Output parameters

□ Convert FASTA program output to big alignment (-bigalign)

Figure 10-4: The fmtseq server.


fmtseq - Windows Internet Explorer
http://bioweb.pasteur.fr/seqanal/interfaces/fmtseq.html
fmtseq
File
Edit
View
Favorites
Tools
Help
Page
Tools
Démarrer
15:21

Input parameters

Set the gap symbol for the input (-gapin) ? ○ [default] ○ '.' ○ '-'

Input sequence format (-inf[ormat])

[Return to the main part with your favorite browser's Back function]

Output parameters

□ convert to lowercase (-c)

□ convert to upper case (-C)

□ Remove gaps from sequences (-degap)

Set the gap symbol for the output (-gapout) ? ○ [default] ○ '.' ○ '-'

□ Only list sequence information (-li)

□ Long form conversion (input header included as comment) (-long)

Gaps in sequences (-raw) ? ○ [default] ○ leave gaps ○ do not leave gaps

□ Reverse-complement each sequence (-re)

Pretty-print options

Figure 10-5: Output parameters of fmtseq.

most cases, you only realize this when it's too late. Table 10-3 lists the kinds of features that formatting can destroy.

Along the same lines, do not take for granted that similar online servers do the same thing (even if they have the same name and the same interface). Two servers running READSEQ may run different versions of this program, or the same version with different default parameters. As a consequence, a problem that doesn't occur with one server may occur with the next server. It really pays to keep your eyes peeled and to keep backup copies of your original files.

Table 10-3 Information You Can Lose When Reformatting

Information Type	*Nature of the Loss*
Sequence name	Long names can be truncated when switching formats. Special characters may also be modified. This happens when converting from FASTA to ALN. The effect of the truncation is unpredictable. Sometimes a portion of the name is added to the sequence!
Upper/lowercase	Case sometimes contains useful information. Strictly speaking, FASTA only supports uppercase. Some programs hate to receive an input with a mixture of cases.
Gap type	Some formats, such as MSF, use different symbols for different types of gaps (. , –, ~). These are often turned into (–) symbols after reformatting.
Annotation	MSF can support weight values for the sequences. This information is lost in most conversions. The extra line of annotation in PIR often disappears when changing the format of these alignments. Any annotation that comes after the sequence name in FASTA is bound to disappear after a conversion.
Special amino acid or nucleotides	Some reformatting programs support the code for ambiguities such as X (for undetermined amino acids) or N (for nucleotides). If your sequences travel across many programs, these symbols may disappear. This can be a problem if you rely on the offset of some residues within the sequences. If you can, stick to the standard 4-nucleotide alphabet and the 20-amino-acid alphabet.

Multiple sequence alignment methods aren't perfect. The people who make them know this, and the people who use them must also consider this fact.

To produce an alignment that you know is right, you can torture a program (and yourself) until it spits out exactly what you want. This may take time and may never lead to the desired result. You can also bite the bullet and edit this alignment yourself. The best recipe for instant insanity is trying to modify your multiple sequence alignment with a standard word processor (such as Microsoft Word or any similar product). It is an experience that has shattered some of the smartest people. The chances of never recovering are high, so don't do it!

Because editing a multiple sequence alignment is so complicated, biologists have developed text editors that are specific for multiple sequence alignment. They make it easy for you to see exactly what's going on. They also make it possible to group sequences so you can realign two subgroups without having to modify the entire alignment. Most of these editors require that you install something on your computer. This should not deter you; some of these programs — Seaview, for instance — are really easy to install on a PC.

However, if you want to stick to your browser, you can use Jalview, a Java applet that you need only load into your Web browser for instant action.

Jalview is a Java applet and uses the same language as Dotlet (see Chapter 8) and has the same general properties:

- It runs on your own computer.
- When you load a sequence in Jalview, your sequence does not travel over the Internet; it stays in your computer.

  If you want to ensure that none of your data travels across the Internet, choose the File➪Work Offline option on your browser as soon as Jalview is loaded.

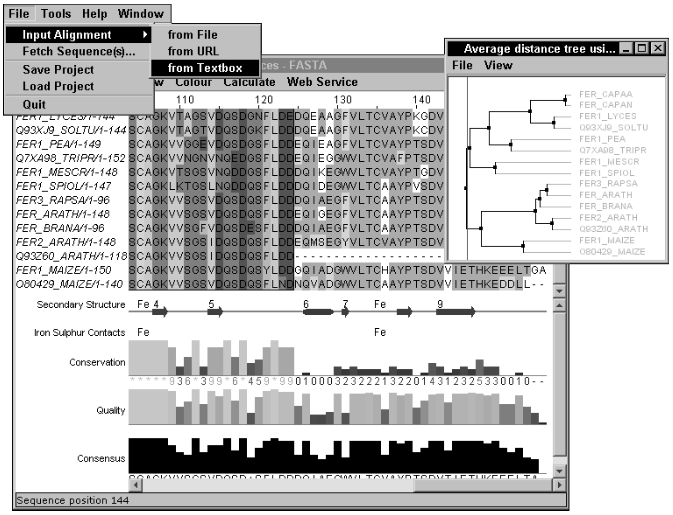
  Do not load confidential sequences in Jalview BEFORE doing this. The Web interface is NOT secure.

www.jalview.org/help.html.

1. **Point your browser to** www.jalview.org/download.html**.**
2. **Click the Install button.**
3. **Accept every proposition prompted to you by the system.**

   If this is the first time you are using Jalview, it will need to upload the java applet. This will take a few minutes. It will then ask you to accept a few permissions having to do with network access — permissions you should graciously grant. If you have already used Jalview before, your computer will only check that you are using the latest version.
4. **Close ALL the windows that appear within the Jalview Window, as they only contain sample data.**
5. **Select the File➪Input Alignment➪From Textbox option, as shown in Figure 10-6.**

   If your alignment is already in a file, you can upload it with the From File option.
6. **Cut and paste your multiple alignment into the text box that appears (Figure 10-7) and click on the Accept box.**



```
FER1_LYCES/1-144     SCAGKVTAGSVDQSDGNFLDEDQEAAGFVLTCVAYPKGDV
Q93XJ9_SOLTU/1-144   SCAGKVTAGTVDQSDGKFLDDDQEAAGFVLTCVAYPKCDV
FER1_PEA/1-149       SCAGKVVGGEVDQSDGSFLDDEQIEAGFVLTCVAYPTSDV
Q7XA98_TRIPR/1-152   SCAGKVVNGNVNQEDGSFLDDEQIEGGWVLTCVAFPTSDV
FER1_MESCR/1-148     SCAGKVTSGSVNQDDGSFLDDDQIKEGWVLTCVAYPTGDV
FER1_SPIOL/1-147     SCAGKLKTGSLNQDDQSFLDDDQIDEGWVLTCAAYPVSDV
FER3_RAPSA/1-96      SCAGKVVSGSVDQSDQSFLDDDQIAEGFVLTCAAYPTSDV
FER_ARATH/1-148      SCAGKVVSGSVDQSDQSFLDDEQIGEGFVLTCAAYPTSDV
FER_BRANA/1-96       SCAGKVVSGFVDQSDESFLDDDQIAEGFVLTCAAYPTSDV
FER2_ARATH/1-148     SCAGKVVSGSIDQSDQSFLDDEQMSEGYVLTCVAYPTSDV
Q93Z60_ARATH/1-118   SCAGKVVSGSIDQSDQSFLDD-----------------
FER1_MAIZE/1-150     SCAGKVVSGSVDQSDQSYLDDGQIADGWVLTCHAYPTSDVVIETHKEEELTGA
O80429_MAIZE/1-140   SCAGKVVSGSVDQSDQSFLNDNQVADGWVLTCAAYPTSDVVIETHKEDDLL--
                     *****936*399*6*459*9901000323222132201431232533001O--
```

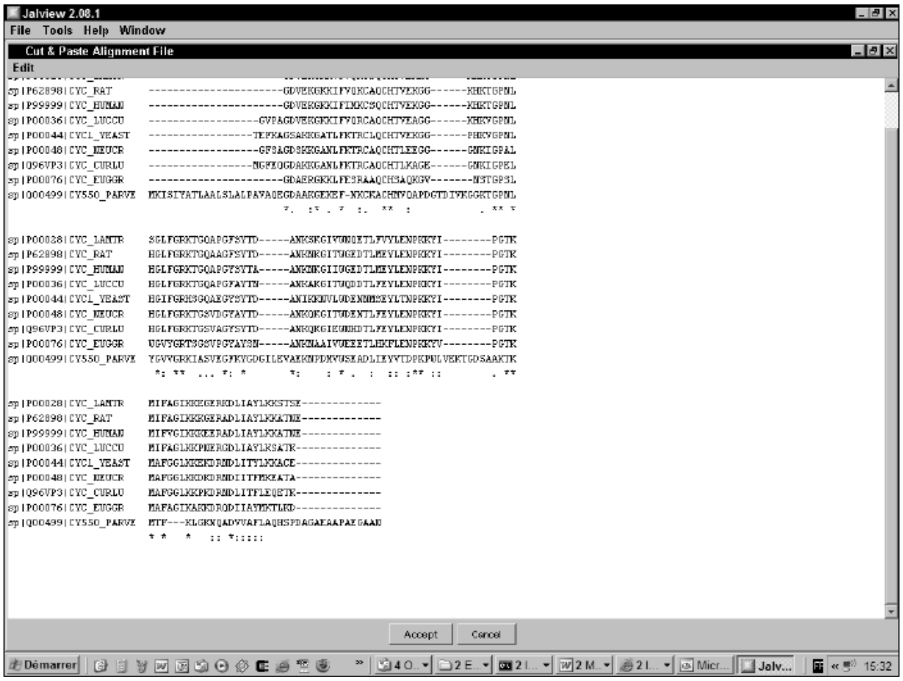
Figure 10-6: Uploading a multiple sequence alignment in Jalview.

7. **Select the ClustalX Color Scheme in the Color pull-down Menu (Figure 10-8).**

   Choosing a color scheme is really a matter of taste; most multiple-sequence-alignment specialists can spend hours explaining to you why they prefer a special color scheme to every other alternative in the galaxy. Whatever your choice, any time you show a color alignment, you must be ready to answer the most common question in the world of biology:

   "What is the color scheme in this alignment?"

   To know more about the color schemes available in Jalview, read the excellent online documentation for this program at `www.jalview.org/help.html`; you'll find an extensive section on color schemes there. In Jalview, all the available color schemes are under the Colour menu (Figure 10-8). The ClustalX, initially designed by Tobby Gibson at the European Molecular Biology Laboratory, is probably one of the most popular.



```
sp|P62898|CYC_RAT          ----------------------GDVEKGKKIFVQKCAQCHTVEKGG------KHKTGPNL
sp|P99999|CYC_HUMAN        ----------------------GDVEKGKKIFIMKCSQCHTVEKGG------KHKTGPNL
sp|P00036|CYC_LUCCU        ------------------GVPAGDVEKGKKIFVQRCAQCHTVEAGG------KHKVGPNL
sp|P00044|CYC1_YEAST       -----------------TEFKAGSAKKGATLFKTRCLQCHTVEKGG------PHKVGPNL
sp|P00048|CYC_NEUCR        ------------------GFSAGDSKKGANLFKTRCAQCHTLEEGG------GNKIGPAL
sp|Q96VP3|CYC_CURLU        -----------------NGFEQGDAKKGANLFKTRCAQCHTLKAGE------GNKIGPEL
sp|P00076|CYC_EUGGR        ----------------------GDAERGKKLFESRAAQCHSAQKGV-------NSTGPSL
sp|Q00499|CY550_PARVE      MKISIYATLAALSLALPAVAQEGDAAKGEKEF-NKCKACHMVQAPDGTDIVKGGKTGPNL
                                                 *.   :* . *  :.   **  :          . ** *

sp|P00028|CYC_LANTR        SGLFGRKTGQAPGFSYTD-----ANKSKGIVWNNETLFVYLENPKKYI--------PGTK
sp|P62898|CYC_RAT          HGLFGRKTGQAAGFSYTD-----ANKNKGITWGEDTLMEYLENPKKYI--------PGTK
sp|P99999|CYC_HUMAN        HGLFGRKTGQAPGYSYTA-----ANKNKGIIWGEDTLMEYLENPKKYI--------PGTK
sp|P00036|CYC_LUCCU        HGLFGRKTGQAPGFAYTN-----ANKAKGITWQDDTLFEYLENPKKYI--------PGTK
sp|P00044|CYC1_YEAST       HGIFGRHSGQAEGYSYTD-----ANIKKNVLWDENNMSEYLTNPKKYI--------PGTK
sp|P00048|CYC_NEUCR        HGLFGRKTGSVDGYAYTD-----ANKQKGITWDENTLFEYLENPKKYI--------PGTK
sp|Q96VP3|CYC_CURLU        HGLFGRKTGSVAGYSYTD-----ANKQKGIEWNHDTLFEYLENPKKYI--------PGTK
sp|P00076|CYC_EUGGR        WGVYGRTSGSVPGYAYSN-----ANKNAAIVWEEETLHKFLENPKKYV--------PGTK
sp|Q00499|CY550_PARVE      YGVVGRKIASVEGFKYGDGILEVAEKNPDMVWSEADLIEYVTDPKPWLVEKTGDSAAKTK
                            *: **  ... *: *        *:     : * .  :  :: :** ::        . **

sp|P00028|CYC_LANTR        MIFAGIKKEGERKDLIAYLKKSTSE-------------
sp|P62898|CYC_RAT          MIFAGIKKKGERADLIAYLKKATNE-------------
sp|P99999|CYC_HUMAN        MIFVGIKKKEERADLIAYLKKATNE-------------
sp|P00036|CYC_LUCCU        MIFAGLKKPNERGDLIAYLKSATK--------------
sp|P00044|CYC1_YEAST       MAFGGLKKEKDRNDLITYLKKACE--------------
sp|P00048|CYC_NEUCR        MAFGGLKKDKDRNDIITFMKEATA--------------
sp|Q96VP3|CYC_CURLU        MAFGGLKKPKDRNDLITFLEQETK--------------
sp|P00076|CYC_EUGGR        MAFAGIKAKKDRQDIIAYMKTLKD--------------
sp|Q00499|CY550_PARVE      MTF---KLGKNQADVVAFLAQHSPDAGAEAAPAEGAAN
                           * *   *   :: *:::::
```

Figure 10-7: The Jalview Alignment TextBox.

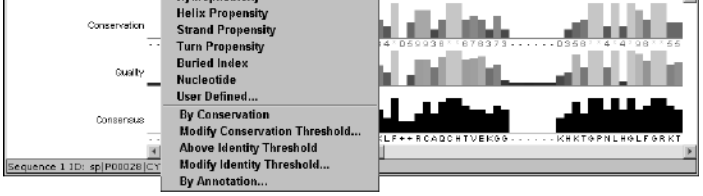


Figure 10-8: Changing the Jalview Color Scheme.

Editing a group of sequences

When you edit an alignment, you usually want to keep some of your sequences aligned the way they are relative to one another. What you want to do is *collectively* modify their alignment. To do this, you need to define them as a group, as follows:

1. **For the purposes of this example, input the dummy alignment we describe in the previous section into Jalview.**

   You can grab the dummy alignment from `www.tcoffee.org/dummy_aln.html`.

2. **Keep the Ctrl key pressed while you click sequences 1, 2, and 4 to select them, as shown in Figure 10-9.**

   Selecting the sequences on a phylogenetic tree is also possible:

   a. Select Calculate➪Calculate Tree ➪Neighbor Joining Tree Using PID.

   b. On the tree that pops up, you can select sequences individually (Figure 10-10).

3. **To insert/remove gaps:**

   a. Keep the Ctrl key pressed.

   b. Put your mouse pointer right where you want to insert or remove the gap.

   c. Drag to the left or to the right to shift your sequences, as shown in Figure 10-11.

   You can edit one sequence at a time by pressing the Shift key instead of Ctrl.

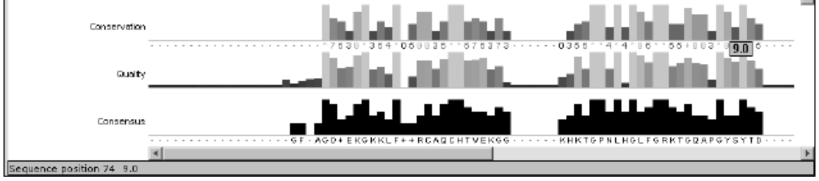


Figure 10-9: Selecting sequences in Jalview.

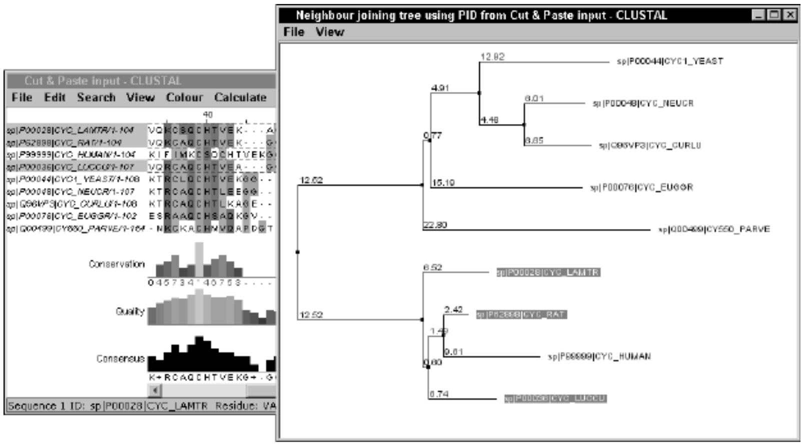


Figure 10-10: Selecting sequences on a phylogenetic tree.

4. **To remove empty columns, use Edit➪Remove empty columns.**

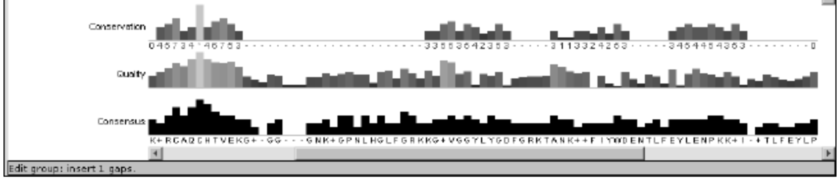
   Empty columns (containing only gaps) may appear while you edit. This simple command gets rid of them for you.

5. **Choose Edit➪Pad Gaps to ensure that no empty column appears at the end of the multiple sequence alignment.**

   When you shift some sequences, you insert gaps. If you don't then edit the gaps at the end of the alignment, you will have problems using your alignment in other applications, as your sequences will seem to be unaligned.

6. **To edit another alignment, go back to the main Jalview window.**

   The File menu in the main window and the File menu in the alignment window do not display the same functionalities.



Edit group: insert 1 gaps.

Figure 10-11: Effect of dragging a group to the right.

Useful features of Jalview

Although explaining all the possibilities of Jalview is clearly beyond the scope of this chapter, we encourage you to explore them. Jalview is like all the programs we introduce here: You can't break anything, so experiment away!

That said, remember that it is easy to ruin a good alignment. Save intermediate results and do not hesitate to use the magic undo button (Ctrl+Z).

Table 10-4 lists some of the features we find especially useful.

Table 10-4 Some Useful Features of Jalview

Command	*Description*
Calculate⇨Autocalculate Consensus	Automatic update of the graph below the alignment. This graph — looking like a city skyline — indicates the level of conservation within the alignment. If you set this option, the graph is updated automatically while you're editing.
Edit⇨Remove Redundancy	Makes sure that no pair of sequences is more than *x* percent similar.
Calculate⇨Tree⇨Neighbor Joining Tree Using PID	Computes and displays a phylogenetic tree in graphic format on which you can select sequences for group editing.

Saving your alignment in Jalview

The new Jalview makes it very easy to save either a colored version of your alignment or a simple text version. We recommend you save both. The colored

1. **From the Jalview Alignment Window, choose File➪Save Alignment AS.**
2. **In the dialog box that appears, select FASTA in the File Type pull-down menu.**

   We recommend you either select the FASTA format (most portable) or the CLUSTAL format (ALN), which is more easily read by us humans.
3. **From the Jalview Alignment Window, choose File➪Export➪HTML.**

   Doing so saves a colorized version of your alignment that you can visualize with any Web browser (Mozilla, Netscape).

Preparing Your Multiple Alignment for Publication

Showtime has finally come: You have the multiple alignment you want, and you're determined to show the world! You want to show this alignment to your colleagues, you want to include it in publications, on posters, in mails, and maybe on T-shirts, mugs, and other merchandise — you're even mulling over the idea of a tattoo. In short, you need a high-impact picture to convince people that your research is going gangbusters. In this section, we list a few online utilities that you can use to beautify your multiple sequence alignments and make them look sharp as a tack.

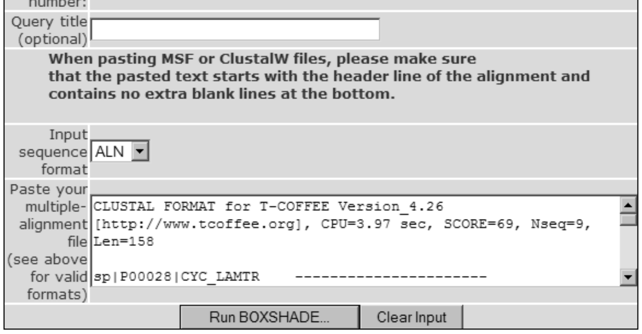
Using Boxshade

Boxshade is a utility that allows you to put some life into your alignment. It shades columns according to their level of conservation and produces files that you can easily manipulate for inclusion in reports or articles. (We give you other powerful tools of the same kind in Table 10-7.)

To get the ball rolling with Boxshade, do the following:

1. **Point your browser to** `www.ch.embnet.org/software/BOX_form.html`**.**

   The Boxshade page of `ch.EMBnet.org` appears, part of which you can see in Figure 10-12.



number:

Query title (optional)

When pasting MSF or ClustalW files, please make sure that the pasted text starts with the header line of the alignment and contains no extra blank lines at the bottom.

Input sequence format: ALN

Paste your multiple-alignment file (see above for valid formats):

```
CLUSTAL FORMAT for T-COFFEE Version_4.26
[http://www.tcoffee.org], CPU=3.97 sec, SCORE=69, Nseq=9,
Len=158

sp|P00028|CYC_LAMTR    ----------------------
```

Figure 10-12: Part of the Boxshade home page.

2. **Choose RTF_new from the Output Format drop-down menu.**

   Most of these formats are difficult to use on a PC. The most convenient is RTF New (Rich Text Format). It generates a file that most word processors (such as Microsoft Word) can read.

3. **Select the font size you want.**

   If your alignment is long, select a small font size. If you have selected RTF_new as your output, you can change your font size later.

4. **Choose Add a Consensus Line with Letters from the Consensus Line drop-down menu.**

   This adds a consensus sequence that contains the most common amino acid for each column.

5. **Select the fraction of sequences you would like shaded.**

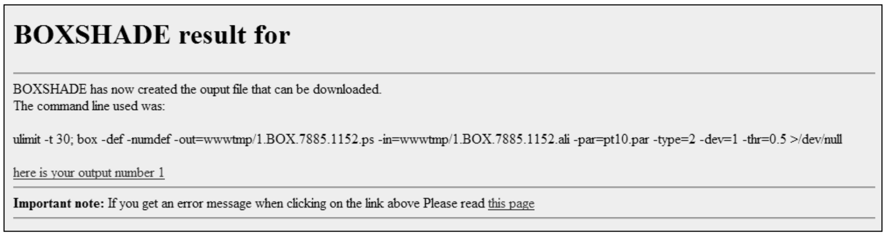
   Boxshade only shades columns that have a certain level of conservation. This feature ensures that conserved columns show up in your alignment. If you request 0.5, for example, it means you want half the amino acids (or nucleotides) to be conserved for some shading to occur.

   Conservation doesn't necessarily mean *identity* in Boxshade. Similar residues, such as isoleucine and valine, also account for conservation. Two types of shading exist:

   - *Black:* Identical amino acids or nucleotides
   - *Gray:* Similar amino acids

7. **Paste your multiple sequence alignment into the Sequence window.**
8. **Click the Run Boxshade button.**

   An intermediate page appears, as shown in Figure 10-13.



BOXSHADE result for

BOXSHADE has now created the ouput file that can be downloaded.
The command line used was:

```
ulimit -t 30; box -def -numdef -out=wwwtmp/1.BOX.7885.1152.ps -in=wwwtmp/1.BOX.7885.1152.ali -par=pt10.par -type=2 -dev=1 -thr=0.5 >/dev/null
```

here is your output number 1

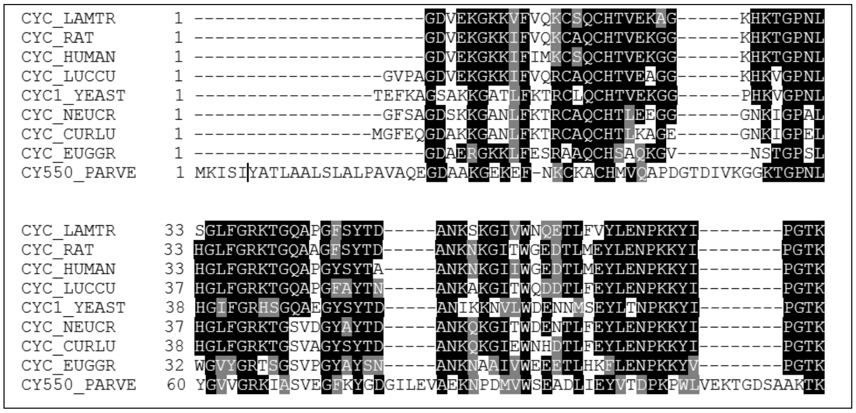
Important note: If you get an error message when clicking on the link above Please read this page

Figure 10-13: The Boxshade intermediate page.

   If this page does not contain an active hyperlink, it means that Boxshade had troubles reading your alignment. (Refer to Figure 10-13 for an example of an active hyperlink.) When this happens, double-check the alignment format. If this doesn't help, use a different format.

9. **Click the** here is your output **link, save to a local file, then open the local file with Word.**

   Figure 10-14 shows you what your multiple sequence alignment looks like after proper shading.



```
CYC_LAMTR      1 ----------------------GDVEKGKKVFVQKCSQCHTVEKAG------KHKTGPNL
CYC_RAT        1 ----------------------GDVEKGKKIFVQKCAQCHTVEKGG------KHKTGPNL
CYC_HUMAN      1 ----------------------GDVEKGKKIFIMKCSQCHTVEKGG------KHKTGPNL
CYC_LUCCU      1 ------------------GVPAGDVEKGKKIFVQRCAQCHTVEAGG------KHKVGPNL
CYC1_YEAST     1 -----------------TEFKAGSAKKGATLFKTRCLQCHTVEKGG------PHKVGPNL
CYC_NEUCR      1 ------------------GFSAGDSKKGANLFKTRCAQCHTLEEGG------GNKIGPAL
CYC_CURLU      1 -----------------MGFEQGDAKKGANLFKTRCAQCHTLKAGE------GNKIGPEL
CYC_EUGGR      1 ----------------------GDAERGKKLFESRAAQCHSAQKGV-------NSTGPSL
CY550_PARVE    1 MKISIYATLAALSLALPAVAQEGDAAKGEKEF-NKCKACHMVQAPDGTDIVKGGKTGPNL


CYC_LAMTR     33 SGLFGRKTGQAPGFSYTD-----ANKSKGIVWNQETLFVYLENPKKYI--------PGTK
CYC_RAT       33 HGLFGRKTGQAAGFSYTD-----ANKNKGITWGEDTLMEYLENPKKYI--------PGTK
CYC_HUMAN     33 HGLFGRKTGQAPGYSYTA-----ANKNKGIIWGEDTLMEYLENPKKYI--------PGTK
CYC_LUCCU     37 HGLFGRKTGQAPGFAYTN-----ANKAKGITWQDDTLFEYLENPKKYI--------PGTK
CYC1_YEAST    38 HGIFGRHSGQAEGYSYTD-----ANIKKNVLWDENNMSEYLTNPKKYI--------PGTK
CYC_NEUCR     37 HGLFGRKTGSVDGYAYTD-----ANKQKGITWDENTLFEYLENPKKYI--------PGTK
CYC_CURLU     38 HGLFGRKTGSVAGYSYTD-----ANKQKGIEWNHDTLFEYLENPKKYI--------PGTK
CYC_EUGGR     32 WGVYGRTSGSVPGYAYSN-----ANKNAAIVWEEETLHKFLENPKKYV--------PGTK
CY550_PARVE   60 YGVVGRKIASVEGFKYGDGILEVAEKNPDMVWSEADLIEYVTDPKPWLVEKTGDSAAKTK
```

Figure 10-14: Boxshade output.

in Figure 10-1. Notice how the conserved amino acids (such as cysteines) stick out, indicating regions of potential biological importance.

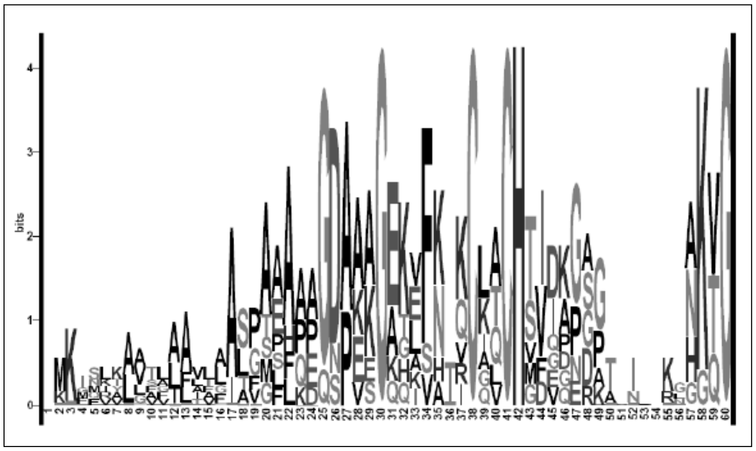


Figure 10-15: Logo representation of a multiple sequence alignment.

When looking at a sequence logo, you can consider the following elements:

- Each position corresponds to a column in the multiple alignment.
- The total height of a logo position depends on the degree of conservation in the corresponding multiple alignment column. Very conserved alignment columns give you high logo positions. Positions that contain a very heterogeneous mixture of symbols yield low logo positions.
- The size of each letter in a logo position depends on how frequent this letter is in the column.
- The top letter is always the most frequent in the column.

It's possible to build logos with either protein or nucleotide multiple sequence alignments. Of course, the better the alignment, the more informative the logo. The most complete logo server around is the one from Berkeley (at `weblogo.berkeley.edu/`) — and it is probably the easiest-to-use server available today. You simply cut and paste your alignment in any convenient format (such as ALN or FASTA) and the server returns a logo like the one in Figure 10-15. A slightly more complicated alternative is also available from: `www.cbs.dtu.dk/~gorodkin/appl/plogo.html`.

Editing and Analyzing Multiple Sequence Alignments for Free over the Internet

This last section briefly introduces and/or recapitulates some lists of resources available on the Internet for editing, analyzing, or visualizing your multiple sequence alignments. If you need to reformat your multiple sequence alignment, you can use the resources indicated in Table 10-2.

Finding multiple-sequence-alignment editors

The only tool available for editing multiple sequence alignments online is Jalview. If you're ready to install some program locally on your machine, a vast selection of high-quality packages exists. Table 10-5 lists some of these programs; most come with extensive documentation for installation and usage.

Table 10-5 Packages for Editing Multiple Sequence Alignments

Name	*Address*	*Description*
Jalview	`www.jalview.org` `www.es.embnet.org/Services/MolBio/jalview/`	Java package, available online
Kalignview	`msa.cgb.ki.se`	Nice online alignment viewer
CINEMA	`www.bioinf.man.ac.uk/dbbrowser/CINEMA2.1/`	A very complete Java package
Seaview	`pbil.univ-lyon1.fr/software/seaview.html`	A beautiful editor, very easy to install
Belvu	`www.cgr.ki.se/cgr/groups/sonnhammer/Belvu.html`	Useful for removing redundancy

(continued)

RALEE	www.sanger.ac.uk/Users/sgj/ralee/	An RNA viewer
Review	bioweb.pasteur.fr/cgi-bin/seqanal/review-edital.pl	A very complete list of viewers

Finding tools to interpret your multiple sequence alignment

The interpretation of a multiple alignment depends very much on its appearance. Some tools on the Net can help you make sense of your multiple alignments by extracting blocks or singling out special positions. Table 10-6 lists some of these resources.

Table 10-6 Extracting Information from a Multiple Sequence Alignment

Name	*Address*	*Description*
Logo	weblogo.berkeley.edu, www-lmmb.ncifcrf.gov/~toms/sequencelogo.html, www.cbs.dtu.dk/~gorodkin/appl/plogo.html	Logos
Blocks	blocks.fhcrc.org/blocks/process_blocks.html	Identifies blocks
Blockgap	www.bork.embl-heidelberg.de/Alignment/blockgap.html	Measures blocks
Lama	blocks.fhcrc.org/blocks-bin/LAMA_search.sh	Compares your multiple alignment with the BLOCKs database
Amas	www.compbio.dundee.ac.uk/servers/amas_server.html	Identifies important features in the multiple alignment

plest tools for improving the visual aspect of a multiple sequence alignment. It isn't the only server you can use for this purpose, however. Table 10-7 lists some alternative resources that are available online.

Table 10-7 Multiple Alignment Beautifying Tools

Name	*Address*	*Description*
ESPript	`espript.ibcp.fr`	A very powerful shading-and-coloring tool
Boxshade	`www.ch.embnet.org/software/BOX_form.html`	Shading in black and white
Mview	`bioweb.pasteur.fr/seqanal/interfaces/mview_blast-simple.html`	Can process BLAST alignments