Supplementary material

Table of contents

Gene analysis and database filling

FAST DB algorithm has been written in PERL (v5.8.5, http://www.**perl**.org/) using the following modules:
- Bioperl (http://www.bio.perl.org/)
- GD (http://www.boutell.com/gd/)
- PDF::API2 (http://search.cpan.org/dist/PDF-API2/)
- Bio::EnsEMBL (http://www.ensembl.org/)
- CGI (http://stein.cshl.org/WWW/software/CGI/)
- DBI (http://dbi.perl.org/)

The FAST DB algorithm is divided in three parts:

- Recovery of data from public databanks
- Data processing to define the FAST DB data set
- Filling of the MySQL database (v4.0.20) (http://www.mysql.com/)

Recovery of data

Data processed by FAST DB algorithm have been retrieved from the following public databanks:
- EnsEMBL (http://www.ensembl.org/) for exonic sequences, genomic sequences and chromosomic position of all genes.
- UCSC (http://genomist.ucsc.edu/) for ESTs, "full length" mRNAs and various information (e.g., tissue expression).
- Genbank (http://www.ncbi.nlm.nih.gov/) for partial mRNAs.
- HUGO and Locuslink (http://www.gene.ucl.ac.uk/nomenclature/) for information relating to gene names and symbols.

EnsEMBL

FAST DB is based on EnsEMBL (version 26, "homo_sapiens_core_26_35"). Exonic sequences, genomic sequences and chromosomic localisations were recovered for all genes. Genomic sequences upstream and downstream the genomic area defined by EnsEMBL for each gene were retrieved simultaneously, to make sure to include additional exons at the gene boundaries. For example, when EnsEMBL sequence is 30 000 nucleotides long, FAST DB will use arbitrarily a sequence of 60,000 nucleotides (15,000 nucleotides at each extremity of the genomic area).

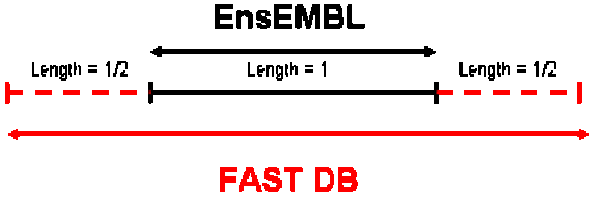


Figure 1: Definition of the genomic area

UCSC

FAST DB uses the assembly 17 of the human genome "human may 2004 (hg17) assembly". mRNAs and ESTs were downloaded together with UCSC tables containing cDNAs-related data. The banks have been formatted by formatdb to be usable by BLAST [(1)] ("formatdb -i downloaded_file -pF -oT -st").

Genbank

Partial mRNAs (others than ESTs) were downloaded from the NCBI website with the following request: "((splic*[Text Word] OR (variant*[Text Word]) OR (isoform*[Text Word])) AND (homo sapiens[Organism]) AND (mRNA[Text Word]) AND (partial[Text Word]) NOT (DNA[Text Word]) NOT (BAC[Text Word]) NOT (contig[Text Word]) NOT (cosmid[Text Word])". This bank of sequences was formatted by formatdb.

HUGO and Locuslink

Data of the HUGO and Locuslink databanks were downloaded and directly inserted into the FAST DB MySQL database.

Data processing

After data had been recovered, FAST DB program analyzed each gene to define the exons and the splicing events. The first step of this process consisted in defining each cluster, that is in selecting all cDNAs corresponding to a given gene.

cDNA selection

For each gene, each exonic sequence was “blasted" against the three banks of sequences ("full length" mRNAs, partial mRNAs and ESTs). For each BLAST output, transcript accession numbers having one E-value lower than 10^{-40} were recovered. The sequence of each pre-selected transcript was recovered (using fastacmd). The selected sequences were aligned against the genomic sequence by sim4 [(2)]. At this point, FAST DB algorithm used five criteria, which are listed below, to eliminate bad quality transcripts or transcripts that are produced by fusion genes associated with diseases (Figure 2).

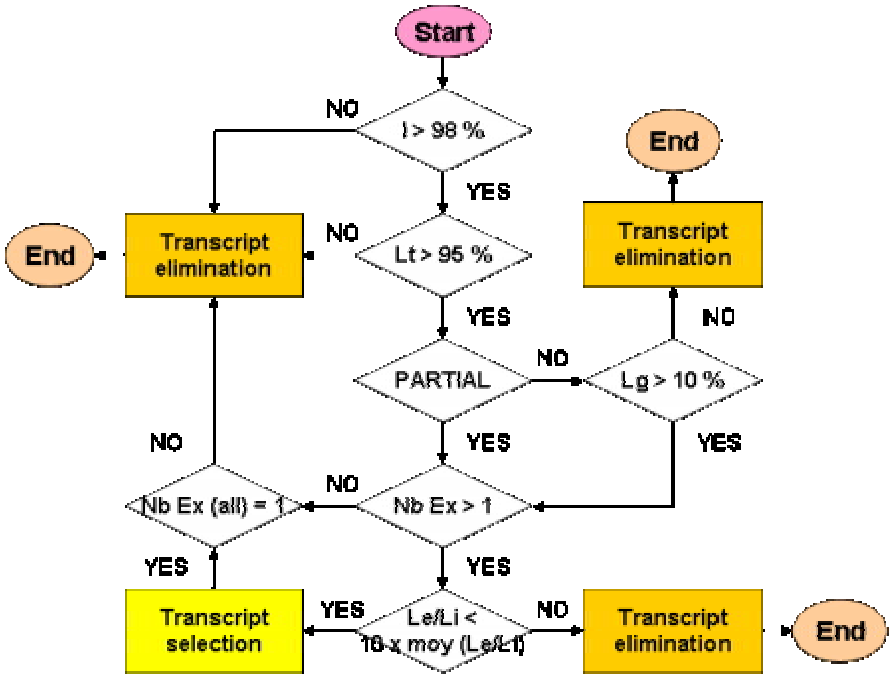
Percentage of identity between genomic and cDNA sequences. To calculate this value for a given transcript, first the percent of identity between each exon and its aligned genomic sequence was normalized by respect to the exon size. The sum of the pondered percents was then divided by the total length covered by the transcript exons (and x 100 to obtain a percentage).

Percentage of cDNA sequence aligned against genomic sequence. This percentage was calculated by dividing the cumulative length of all exons defined by a transcript by the length of the transcript (and x 100 to obtain a percentage).

Number of exons defined by the sim4 alignment. If only one exon was defined, the transcript was excluded unless all the transcripts agreed to define one single exon. In this case, the gene was defined as an “intron-less gene”.

The percentage of the genomic sequence covered by a transcript. This percentage was calculated by dividing *the length of the genomic sequence covered by the alignment* by the length of the gene (and x 100 to obtain a percentage). This value was used to exclude transcripts of fusion genes. Noteworthy, the *length of genomic sequence covered by the alignment* corresponds to the whole genomic sequence localized between the first position in the genomic sequence of the first exon and the last position in the genomic sequence of the last exon defined by the transcripts.

The ratio of the cumulative length of exonic sequences over the cumulative length of intronic sequences. This value was used to exclude cDNAs that had not been spliced through most of their length and that in most cases come from genomic contamination or pre-mRNAs.




```
I           : percentage of identity of cDNA
Lt          : percentage of transcript sequence aligned against genomic sequence
PARTIAL     : partial mRNAs and ESTs
Lg          : percentage of genomic sequence covered by alignment
Nb ex       : number of transcript exon(s) defined by alignment
Nb Ex (all): if all transcripts have the same "Nb Ex" value, this value is labelled "Nb Ex (all)"
Le/Li       : cumulative length of exonic sequence/ cumulative length of intronic sequence
Moy (Le/Li): average of Le/Li (average on all the gene transcripts)
```

Figure 2: Transcript selection algorithm

Definition of Genomic Exons

After the transcripts corresponding to one gene had been selected, the transcript sequences were aligned against the genomic sequence by sim4 [(2)] to define the exon positions on the transcript sequence and on the genomic sequence. We have defined a "genomic exon" as the most frequent exon among all transcript exons at a given genomic position (Figure 3). To do this analysis, all transcript exons were sorted by ascendant order with respect to their first

position in the genomic sequence. They were then gathered by "bag", each "bag" corresponding to one "genomic exon". A transcript exon belongs to the next "bag" (or next genomic exon) when it begins at a genomic position that is at least 30 nts downstream the end of the previous genomic exon (minimum intron length fixed, see Table 1).

Beginning position of exons	***End position of exons***	***Most frequent end of the preceding "genomic exon"***	***Number of the "genomic exon"***
3568	3755	3755	**1**
3568	3755	3755	**1**
3703	3755	3755	**1**
5149	5279	5279	**2**
5149	5279	5279	**2**
5200	5279	5279	**2**
5692	5839	5839	**3**

Figure 3: Genomic exon definition (1/2)

We defined the first and the last positions of a genomic exon as the most frequent first and last positions occupied by the transcript exons belonging to this genomic exon "bag" (see Figure 3). However, the first position of the first exon and the last position of the last exon of the gene were defined differently. The first position of the first exon was defined as the lowest position among the transcript exons. The end position of the last exon was defined as the highest position among the transcript exons. In case of "intron-less" gene, the first position of the single exon was defined as the lowest position and the last position was defined as the highest position.

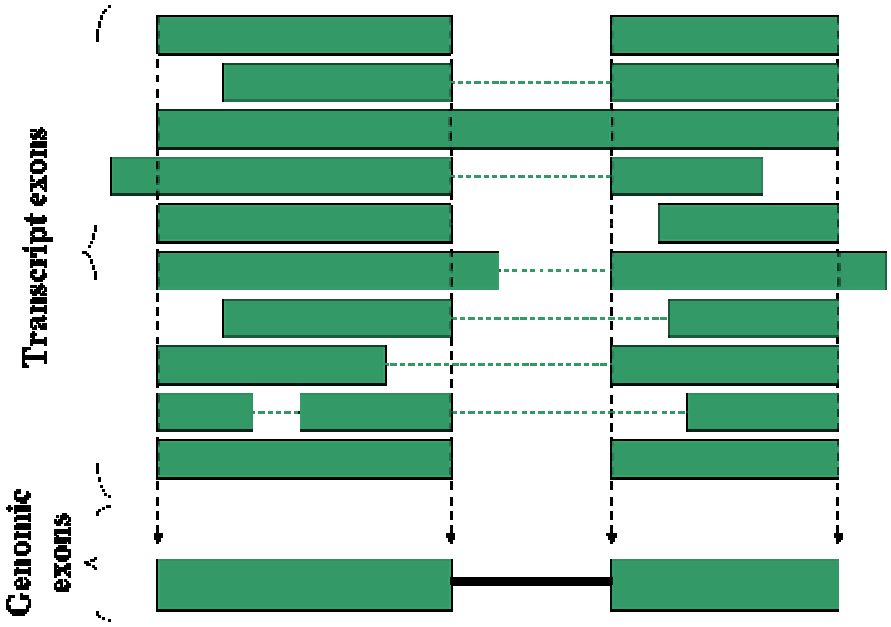


Figure 4: Genomic exon definition (2/2)

Definition of alternative splicing events

After genomic exons have been defined, alternative events were defined by comparing transcript exons with the corresponding genomic exons. Seven types of alternative events were defined:

- Alternative first exon
- Alternative last exon
- Retained intron
- Exon skipping
- Alternative 3' splicing site
- Alternative 5' splicing site
- IED (Internal Exon Deletion): in most cases, this event applies to small introns that are rarely eliminated.

Alternative first exon

To define a transcript exon as an alternative first exon of a gene, we used several criteria. First, the exon must be the first exon of one or more transcripts. Second, if the first exon of a transcript does not co-localize with any genomic exon (case 1, transcript 2 in Figure 5), it is defined as a first exon. If on the contrary, the first exon of a transcript lies within the genomic sequence corresponding to a genomic exon, it is defined as a first exon only if it starts at least 10 bp upstream the first position of the corresponding genomic exon (case 2, transcript 2' in Figure 5); this criterion prevents from defining "false" first exon due to sequencing errors. We performed an additional analysis to eliminate potential false "first exons" defined by the case illustrated in Figure 5, transcript 2'. The upstream sequence of all these exons was aligned against the corresponding genomic sequence and we found that about 4% of these first exons contained a 5'-sequence that did not correspond to the right genomic sequence. The database has been manually corrected by deleting these exons from the "first exon" table.

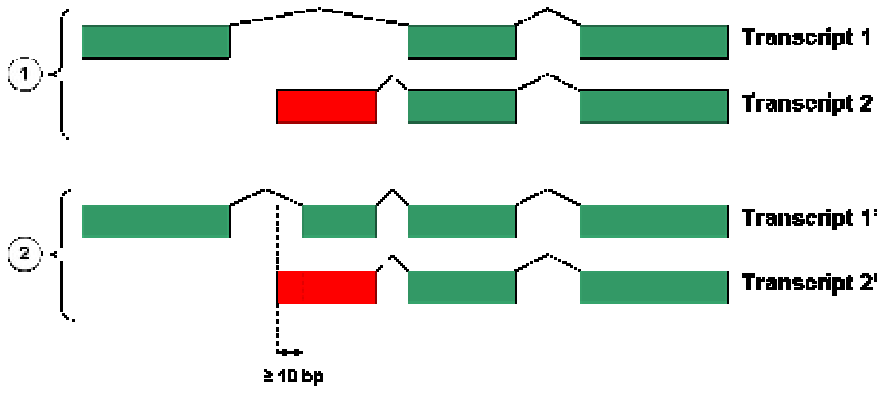


Figure 5: Alternative first exon definition

It is important to underline that one can not exclude that upstream exon(s) that is (are) not present in the available transcript(s), exist. To gain more confidence in a "first exon" candidate, users can perform computer-assisted analyses for promoter and transcription factor binding site prediction and for 5'-UTR features using the candidate "first exon" and upstream sequences and FAST DB interface (see below).

Alternative last exon

Alternative last exons (other than the last exon) are shown in red color in the example in Figure 6. Last exons are defined as being the last exon of at least one transcript and exceeding by at least 10 bases the last position of the corresponding genomic exon (transcript 2' in Figure 6). If there are only last transcript exons at this position, the exon is always considered as an alternative last exon (transcript 2 in Figure 6). We have decided to include more than 10 nucleotides in the next intron to limit the number of false positives due to sequencing errors. We performed an additional analysis to eliminate potential false "last exons" defined by the case illustrated in Figure 6, transcript 2'. The downstream sequence of all these exons was aligned against the corresponding genomic sequence and we found that 0.2% of these last exons contained a 3'-sequence that did not correspond to the right genomic sequence. The database has been manually corrected by deleting these exons from the "last exon" table.

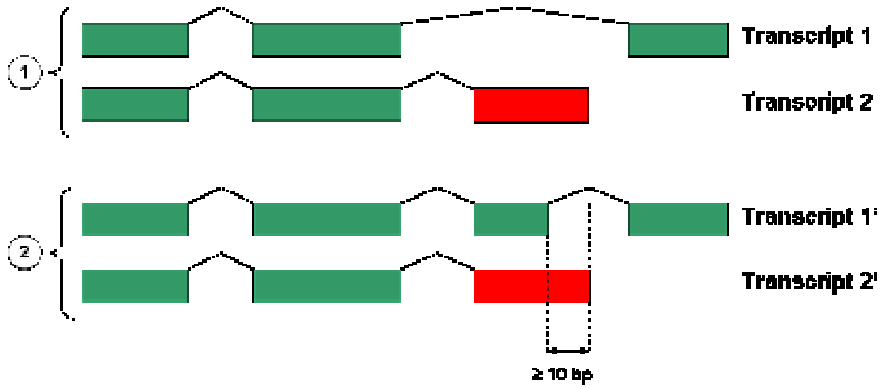


Figure 6: Alternative last exon definition

It is important to underline that one can not exclude that downstream exon(s) that is (are) not present in the available transcript(s), exist. To gain confidence in a "last exon" candidate, users can perform promoter-assisted analyses for polyadenylation site prediction and 3'-UTR features using the candidate "last exon" and downstream sequences and the FAST DB interface (see below).

Retained intron

A transcript exon that ends downstream the first position of the next genomic exon is defined as a retained intron (red exon of transcript 2 in Figure 7). A retained intron may "cover" several consecutive introns. To avoid "false retained introns" in FAST DB coming from genomic or pre-mRNA contamination in the transcript databases, we introduced an optimized criterion during the data processing, which is the ratio of intron length to exon length (see cDNA selection).

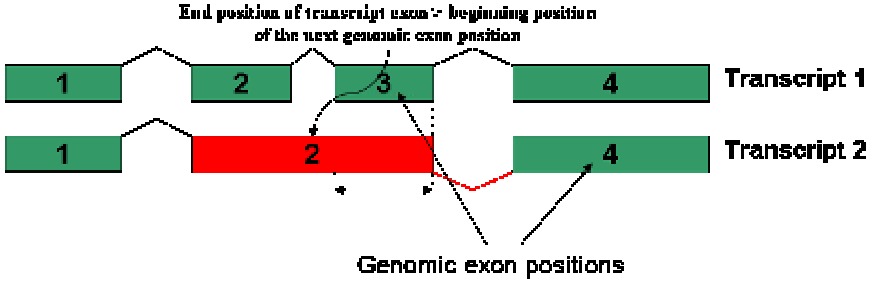


Figure 7: Retained intron definition

Exon skipping

We defined an exon skipping event when two consecutive exons from the same transcript present the following criteria : the first exon is not a retained intron; the genomic position of the second exon is higher than the genomic position of the first exon plus one: in Figure 8, the second exon of transcript 2 is at position three, which is higher than position two (position of the first exon plus one).

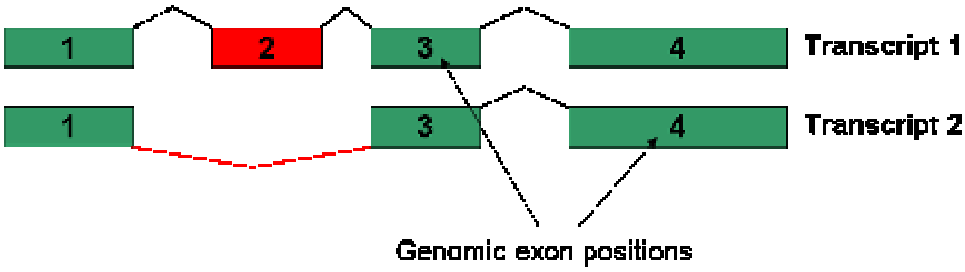


Figure 8: Exon skipping definition

Alternative 3' splicing site

An alternative 3' splicing site was defined when a transcript exon starts at least 3 nucleotides downstream or upstream of the start of the corresponding genomic exon (red exon of transcript 2 in Figure 9). This value was fixed to exclude false positives due to sequencing errors. To exclude false positives due to sim4 alignment problems, another criterion was that the previous transcript exon (from the same transcript) had no alternative 5' splicing site with the sign of its difference of length opposite to the sign of the defined alternative 3' splicing site (orange exons of transcript 3 in Figure 9). Such events seem to be due to sim4 alignment problems in many situations. Finally, to define an alternative 3' splicing site, a transcript exon can not be the first exon of its transcript as first exons do not follow introns.

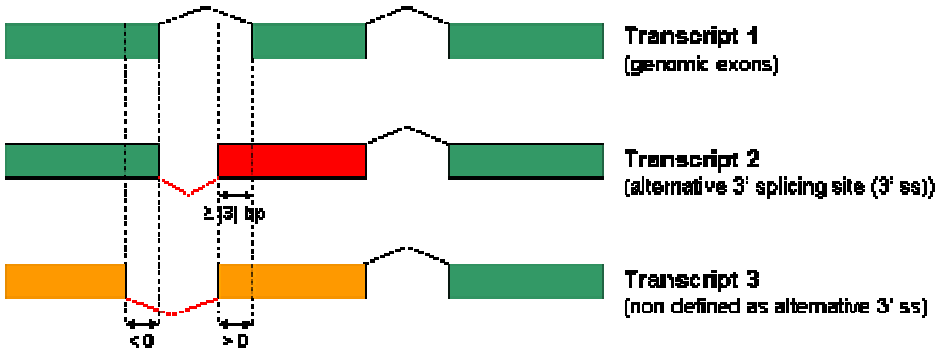


Figure 9: Alternative 3' splicing site definition

Alternative 5' splicing site

An alternative 5' splicing site was defined when a transcript exon ended at least 3 nucleotides downstream or upstream of the end of the corresponding genomic exon (red exon of transcript 2 in Figure 10). This value was fixed to exclude any false positives due to sequencing errors. To exclude false positives due to sim4 alignment problems, another criterion was that the next transcript exon (from the same transcript) had no alternative 3' splicing site with the sign of its difference of length opposite to the sign of the defined alternative 5' splicing site (orange exons of transcript 3 in Figure 10). Indeed, such events seem to be due to sim4 alignment problems in many situations. Finally, to define an alternative 5' splicing site within a transcript exon, this exon can not be the last exon of its transcript as last exons are not followed by introns.

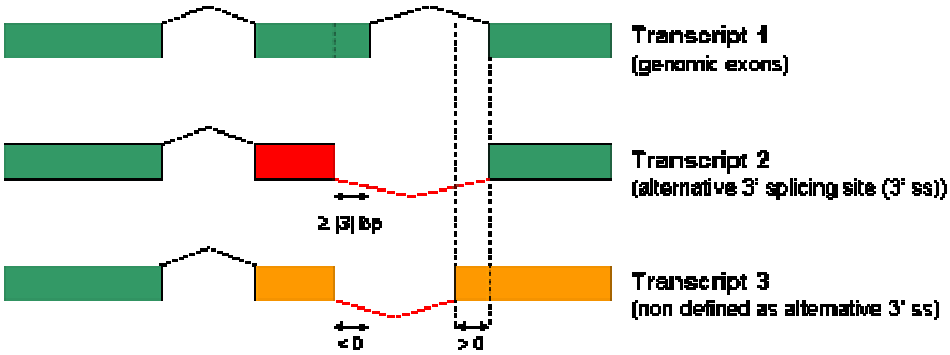


Figure 10: Alternative 5' splicing site definition

Internal Exon Deletion (IED)

An internal exon deletion was defined if at least two transcript exons from the same transcript were at the same genomic exon position and if the length between these exons was equal to or higher than ten nucleotides (red exons of transcript 2 in Figure 11). The limit was fixed at 10 nucleotides to avoid deletions owing to sequencing errors. In most cases, these events seem to correspond to small introns that are rarely eliminated. Indeed, these sequences had consensus acceptor/donor splicing sites (data not shown).

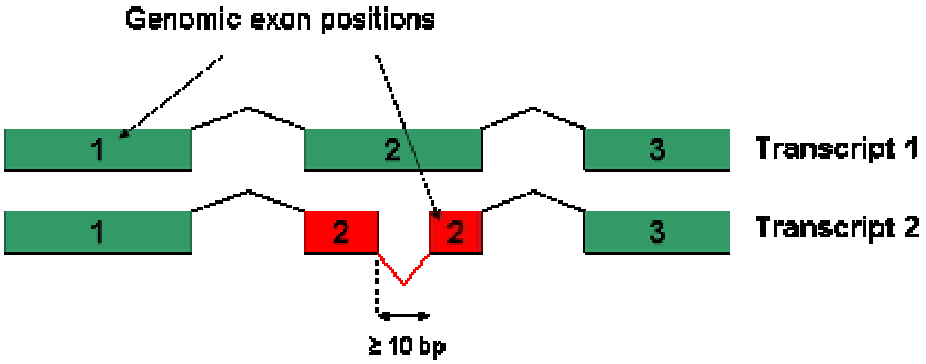


Figure 11: IED definition

Filling the database and creation of PNG and PDF files

Filling the database

After splicing events had been defined, all the results were stored in a MySQL database by requesting insertions.

Creation of PDF files

The program (module Perl PDF::API2) dynamically generated PDF files corresponding to each gene. These files were stored on the hard disk of the server in order to increase the downloading rate for users.

Creation of PNG files

Graphical representations of each gene and its transcripts (Figure 12) were dynamically generated (module GD). The legend of these graphs is described in FAST DB USER's Guide. It is important to underline that on the gene graphical representation, the exons (represented by green rectangles) do not correspond to "genomic exons". A given "genomic exon" is the most frequent exon found within the gene transcripts, whereas a "graphical exon" correspond to the longest exon at this genomic position. In other words, the beginning and the end of each "graphical exon" were defined by the first and last positions occupied by the "transcript exons" at this genomic position.
A black V-shaped line connecting two consecutive genomic exons represents a splicing event between two genomic exons. All other splicing events are represented by a red V-shaped line the exons. It is important to keep in mind that a "black connection" does not correspond to the most frequent splicing event: for example, even if the skipping of exon 3 is the most frequent splicing event, it is represented by a "red connection".
Noteworthy, some volunteer discrepancies may be observed between the "alternative splicing" graphical representation (red lines) and the "alternative splicing" events defined by FAST DB. One example is illustrated in Figure 12. The orange exon of transcript 2 had two additional nucleotides at its 3'-end compared to the most frequent exon end. In this case, the graphical representation shows a red line under the exon even if FAST DB has not defined an alternative 5' splicing site (at least 3 bases would be required to define such an event, see above). This configuration was used to bring the attention of users to additional potential alternative splicing events to which FAST DB does not grant enough confidence.

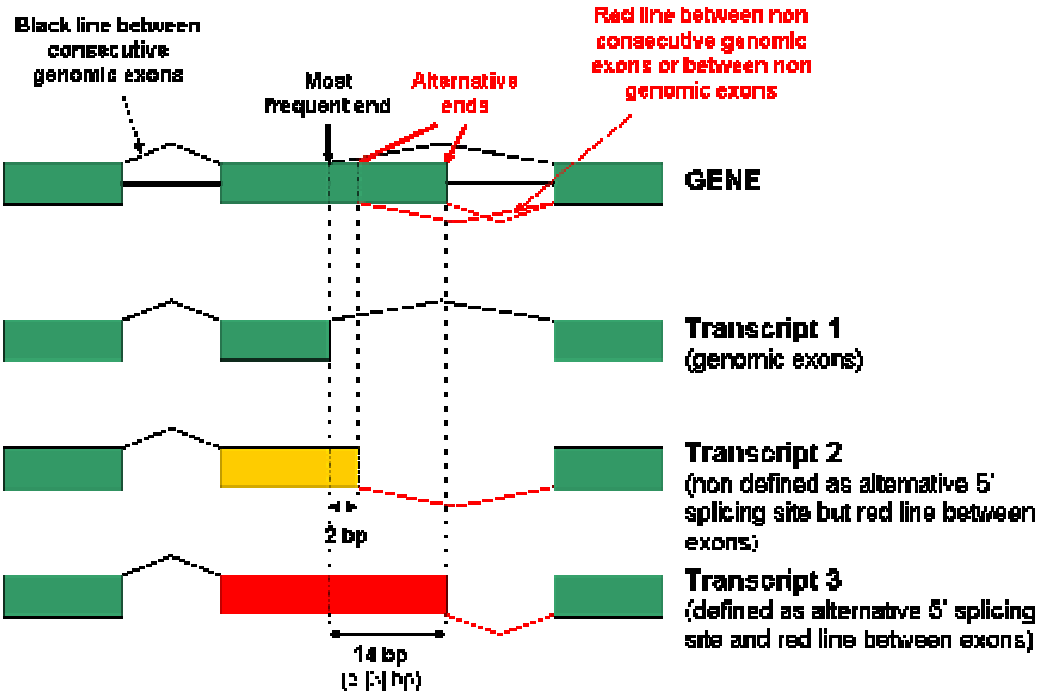


Figure 12: Gene graphical representation (1/2)

In case of a retained intron (red exon of the third transcript in Figure 13), the connection between the exon containing the retained intron and the next exon is represented by a red line.

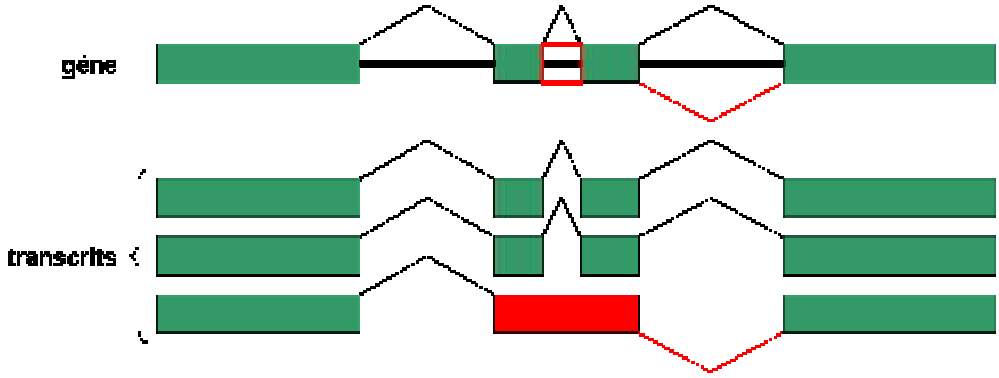


Figure 13: Gene graphical representation (2/2)

Database interface

The FAST DB interface was created in PERL on an APACHE server (http://www.apache.org/). We used the CGI PERL module. This interface has several functions:

- Use the search engine to find a gene
- Clear presentation data
- Navigate through the website
- Direct links to other websites
- Use tools, in particular multiple alignment of transcript sequences and PCR *in silico*

Search engine

The searching engine was designed to retrieve a given gene by two ways: by typing a keyword or by entering a sequence. In the first case, the search engine accepts different attributes:

- Gene name
- Symbols
- Synonyms
- OMIM number
- Locuslink ID
- Refseq accession
- Swissprot accession
- Transcript accession number
- EnsEMBL stable ID

In the second case, the search using a sequence is done by BLAST. The user's input sequence (at least 20 nucleotides) is "blasted" against all the FAST DB genomic sequences.

A multiple query interface is also available. Users might upload a file (rtf, doc, txt…) with EnsEMBL stable ID of several genes. The interface accepts only one EnsEMBL stable ID per line. The FAST DB search engine provides the list of the corresponding genes (each gene is clickable for analysis) and the query result might be saved as an html file for further analysis.

Multiple alignment and in silico PCR

Multi-alignment

The FAST DB program uses Clustalw [3] and Partial Order Alignment with the global alignment option (POA) [4, 5] to perform the multi-alignment of all the transcript sequences of a given gene. However, to avoid mistakes in the alignment due to the amount of sequences to align and to the great differences between them, the FAST DB program "prepares" the sequences to align. First, all transcript exons localized at the same genomic position are identified. In case of a retained intron, the program separates the corresponding transcript exon into several exons and intron(s) (in red color in Figure 15) and FAST DB recovers the

longest genomic exon for each position, not the genomic exon. In the next step, all exons localized at the same genomic position are aligned. Please note that when the same intron retention event is represented by two or more transcripts, FAST DB does not align the intronic sequences, although they are displayed. In case of a genomic position with an IED, FAST DB uses POA. In all the others cases, FAST DB uses Clustalw.

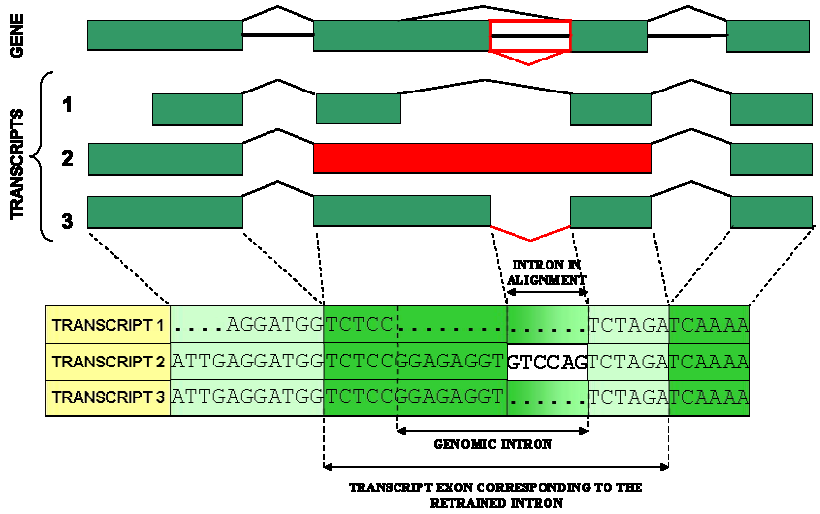


Figure 14: Multi- alignment with retained intron

In some cases, misalignments of sequences occur. The Clustalw program always tries to establish an optimal alignment of a set of sequences by making them start at the same position, which yields wrong alignments when the program tries to align exons that start at very different positions at the same genomic exon (Figure 16, left panel). POA program uses a global alignment option in the case of IED in the first or the last exon (where exons start and end at very different genomic positions), and it does not always properly separate the transcript exons on the multiple alignment (Figure 16, right panel).

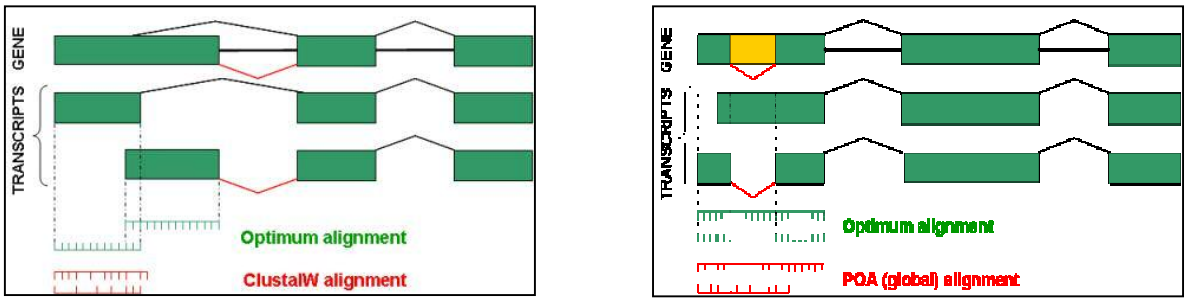


Figure 15: Problems of multi-alignment

To detect any problem in the transcript multi-alignment, the alignment of the transcript sequences against the genomic sequence is provided for each transcript (using SIBsim4) (left panel in Figure 16).

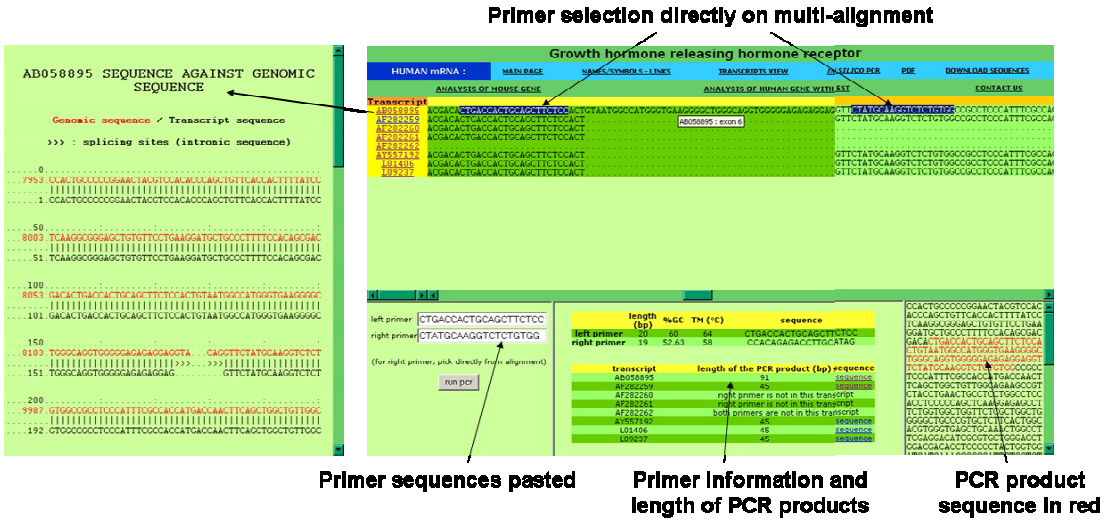


Figure 16: Multi-alignment

In silico PCR

Users define PCR primers directly on the multi-alignment by copying the sequences and pasting them in the corresponding boxes (Figure 16). After clicking the “run PCR” button, information concerning the primers is provided and a table gives the length of the predicted PCR product for each transcript. Please note that the primer sequence must be identical to the transcript sequence in order to get a result: no mismatch is allowed.

Bibliographical references

1. Altschul, Stephen F., Gish, Warren, Miller, Webb, Myers, Eugene W., and Lipman, David J. (1990). Local BASIC alignment search tool. J. Mol. Biol. **215** ; 403-410.

2. L. Florea, G. Hartzell, Z. Zhang, G. Rubin, and W. Miller (1998). With computer program for aligning has DNA sequence with has genomic DNA sequence. Genome Research **8**, 967-974

3. Higgins D., Thompson J., Gibson T.Thompson J.D., Higgins D.G., Gibson T.J.(1994). CLUSTAL W: improving the sensitivity of progressivemultiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. Nucleic Acids LMBO. **22**:4673-4680.

4. Lee C., Grasso C., Sharlow M. (2002). Multiple Sequence Alignment Using Partial Order Graphs. Bioinformatics **18**:452-464.

5. Catherine Grasso C., Lee C. (2004). Combining Partial Order Alignment and Progressive Multiple Sequence Alignment Increases Alignment Speed and Scalability to Very Large Alignment Problems. Bioinformatics 2004 **20**:1546-1556.

Figure index