 Nucleic Acids Research, 2000, Vol. 28, No. 1 **185–190**

EID: the Exon–Intron Database—an exhaustive database of protein-coding intron-containing genes

Serge Saxonov, Iraj Daizadeh, Alexei Fedorov and Walter Gilbert*

Department of Molecular and Cellular Biology, Harvard University, Cambridge, MA 02138, USA

Received August 23, 1999; Revised and Accepted October 25, 1999

ABSTRACT

To aid studies of molecular evolution and to assist in gene prediction research, we have constructed an Exon–Intron Database (EID) in FASTA format. Currently, the database is derived from GenBank release 112, and it contains 51 289 protein-coding genes (287 209 exons) that harbor introns, along with extensive descriptions of each gene and its DNA and protein sequences, as well as splice motif information. There is 17% redundancy inherited from GenBank—a purge at the 99% identity level reduced the database to 42 460 genes (243 589 exons). We have created subdatabases of genes whose intron positions have been experimentally determined. One such database, constructed by comparing genomic and mRNA sequences, contains 11 242 genes (62 474 exons). A larger database of 22 196 genes (105 595 exons) was constructed by selecting on keywords to eliminate computer-predicted genes. By examining the two nucleotides adjacent to the intron boundary, we infer that there is a 2% rate of errors or other deviations from the standard GT…AG motif in nuclear genes. This criterion can be used to eliminate 4921 genes from the overall database. Various tools are provided to enable generation of user-specific subsets of the EID. The EID distribution can be obtained from http://mcb.harvard.edu/gilbert/EID

INTRODUCTION

The intron–exon organization of eukaryotic genes is an intensely studied field of biology (1–13). The origin of introns remains a mystery, and at least one theory—'The Exon Theory of Genes'—links their presence with the origin of genes (3). Thus, questions in molecular evolution are being investigated through *in silico* analysis of intron–exon structures in various organisms. To facilitate such studies, while taking advantage of the exploding amount of sequence data now available, we present an Exon–Intron Database (EID), which harbors 51 289 protein-coding genes that were extracted from the eukaryotic subset of GenBank (release 112) (14). Besides its applications to molecular evolution, the database may be of use to researchers who are exploring ways to improve the accuracy of gene prediction programs. Since the EID provides a well-organized, extensive set of data for studying features of introns and exons, including an ability to focus on experimentally determined data, it is possible that it will lead to new insights that will improve intron/exon recognition.

We provide a flat file distribution of the EID enabling the users to perform large-scale analyses of the features in the database. For each gene, the EID provides the protein sequence, the DNA sequence as well as an extensive description taken from GenBank header information. All the sequences are in FASTA format, simplifying their use in many applications. We also provide intron phase information, intron positions, lengths of introns and exons, along with four-letter strings that contain the first and the last two bases of each intron (which can be used in error-correction, as we discuss below). All of these features are accessible through easily parseable handles.

When examining intron evolution, it is of importance to know the correct positions of introns. Mistakes in the data, even if rare, will have disproportionally high effects in any analysis that involves mappings of intron positions from homologous proteins onto a single sequence (12), in investigations of intron sliding (15), as well as in other studies. Unfortunately, mistakes do occur in GenBank.

The first category is mistakes that are due to clerical errors. In these cases, the intron–exon boundaries are specified incorrectly, usually quite close to the actual boundaries. To identify many of these aberrations, we examine the first two and the last two nucleotides of the intron. The pattern characterized by 'GT…AG' is almost an invariable consensus for spliceosomal introns. In genes subject to typographical errors, the splicing motifs will most likely be other than GT…AG. Therefore, the user can remove all the genes whose introns do not have the canonical GT…AG motif. However, this technique is only valid for spliceosomal introns, as group I and group II introns do not exhibit this strong consensus. Furthermore, there is a recently identified consensus motif 'AT…AC' (16). Only 0.0176% of the introns in our database have this pattern. Finally, it has been suggested that possibly 1% of non-canonical spliceosomal introns are present in *Arabidopsis thaliana* (17).

The second class of incorrectly-identified intron positions arises as a result of inaccurate gene predictions by computer programs. Many of the genes in GenBank have not been confirmed experimentally and thus may be unreliable. This fact is especially troublesome if one is to use the data in the database to improve the techniques of exon/intron recognition. To tackle this problem we used two filtering methods to create subsets of the EID with minimal presence of non-experimentally determined genes. The first approach involved comparing all the genes in the EID against the set of mRNA sequences in GenBank. Thus, if the coding sequence of a particular gene

*To whom correspondence should be addressed. Tel: +1 617 495 0760; Fax: +1 617 496 4313; Email: gilbert@nucleus.harvard.edu

matched a known mRNA sequence we would have great confidence that the gene structure has been established correctly. The resulting subdatabase was 22% of the original EID. This rather low figure can be due to the fact that only a fraction of known mRNA sequences is present in GenBank and is labeled as such. The second filtering method that we used was keyword-based. We parsed the GenBank header information for each gene to discard the genes whose structures were likely to have been predicted. The resulting subset contained 44% of the original database. Even though we believe the keyword-based method works reasonably well, this subset is intrinsically less pure than the one obtained through mRNA-matching, because of its reliance on the often ambiguous GenBank header information. Therefore, if the users need a database of experimentally-confirmed genes, they may choose between a large EID subset with possible minor contamination, and a smaller, but more pure one.

DETAILS OF THE CONSTRUCTION AND THE FORMAT OF THE EXON–INTRON DATABASE

The core of the EID is composed of three files: 'gb112.dEID', 'gb112.pEID' and 'gb112.hEID'. These contain the DNA sequences, the protein sequences and the GenBank header information, respectively. To construct these files we parsed the following .seq files from GenBank (release 112): gbinv1.seq, gbinv2.seq, gbmam.seq, gbpln1.seq, gbpln2.seq, gbpri1.seq, gbpri2.seq, gbpri3.seq, gbpri4.seq, gbrod.seq, gbvrt.seq. We extracted all protein-coding genes (or gene fragments) that exhibited tessellated structures. Isolating such genes from GenBank entailed searching for 'CDS join' features that were marked by one of the following strings:

```
CDS             join
CDS             complement(join
```

The information contained within these features, in principle, specifies the exact positions of exons within the given nucleotide sequences. Furthermore, each CDS feature should contain the protein translation of the coding regions. In our analysis if the translation was absent or was different by more than one residue from the total length of all the exons, we discarded the gene. There were 733 such genes. It bears mentioning that our approach missed a small number of intron-containing genes that were submitted before the adoption of the CDS join feature. Because such entries were rather poorly standardized and because their number was small, we chose to exclude them from our analysis.

To assemble the DNA portion of the EID, we parsed each CDS join feature to extract DNA sequences that were either stored in the given entry or referenced elsewhere. Our parser further took into account possible 'complement' tags within the feature. Obtaining DNA sequences was more straightforward for exons, as it simply required reading pieces of DNA specified in the CDS join features. The determination of intron sequences was slightly more involved as the program had to read how the surrounding exons were being referenced. Thus for a particular intron, if both surrounding exons were in the same GenBank entry and on the same strand (either both or neither were marked with 'complement') the program read the sequence between the two exons to obtain the intron. Sometimes, when the exons were in different entries in GenBank, on different strands in the same entry, or came from an mRNA sequence, the intron sequence could not be determined. In such cases, the program still attempted to look at the DNA surrounding the exons to deduce the splicing motif of the intron. It should be noted that the approach of looking exclusively at the CDS join feature precluded us from obtaining parts of the gene that resided outside the protein-coding region. In general, the 3′ and the 5′ UTR information is much less standardized than the CDS join feature, making its compilation considerably less reliable.

The DNA database was constructed in the standard FASTA format. Thus, each gene has its own entry, where the first line of the entry starts with a '>' sign and contains information about the sequence. This description contains the following list of identifiers: a unique EID index, the GenBank locus, the GenBank protein identifier (protein_id) and a short description extracted from the DEFINITION line of the GenBank entry. In addition, the first line includes the following sequence-specific information: intron phases (positions of introns within codons—could be 0, 1 or 2), intron lengths, exon lengths, the total exon length and the total intron length. If for any of the reasons mentioned above, the intron size is unknown, the letter 'u' is used instead of the actual intron size. Also, if the length of any one of the introns is unknown, the total length is automatically set to –1, denoting that the total length is unknown. The last field of the line contains the splice motif information. Here, for each intron we provide a four-letter string that is composed of the first two and the last two nucleotides of the intron. Sometimes, when only the mRNA sequence is given, or the particular intron simply has not been deposited to GenBank, we are unable to determine the sequence at the start or the end of the intron, and we place 'NN' in the corresponding portion of the motif string. The other case, where we cannot determine the splicing motifs, usually suggests a typographical error. It arises when the distance between neighboring exons is less than four nucleotides. Here, we label the entire motif as 'EEEE'. The nucleotide sequence is presented after the first line. Exons are given in capital letters, while the introns are set to lower case. If the intron sequence is unknown, it is represented by a pair of periods '..'.

The protein part of the EID is composed in the standard FASTA format as well. To construct it, we took translations from every 'CDS join' feature and determined, based on exon lengths, where the introns were relative to the protein sequence. Thus if the codon for a particular residue was interrupted by an intron or was immediately preceded by an intron, the letter for that residue was set in lower case. Similar to the DNA database the first line of each entry contains a series of identifiers: the EID index, the GenBank locus, the protein_id identifier and a short description from the DEFINITION line. This information is followed by lists of intron positions, intron phases and exon sizes, along with the total length of the protein. All the positions and lengths are given in terms of residues. The rest of the entry consists of the protein sequence.

The third file mentioned above contains extensive descriptive information for all the sequences given in the other two databases. The entries in all three databases follow the same indexing, facilitating their concurrent use. The description for each entry in the third database was extracted from the header portion of the corresponding GenBank entry, and includes both the information about the locus as well as the information specific to the gene. These points are illustrated in Figure 1.

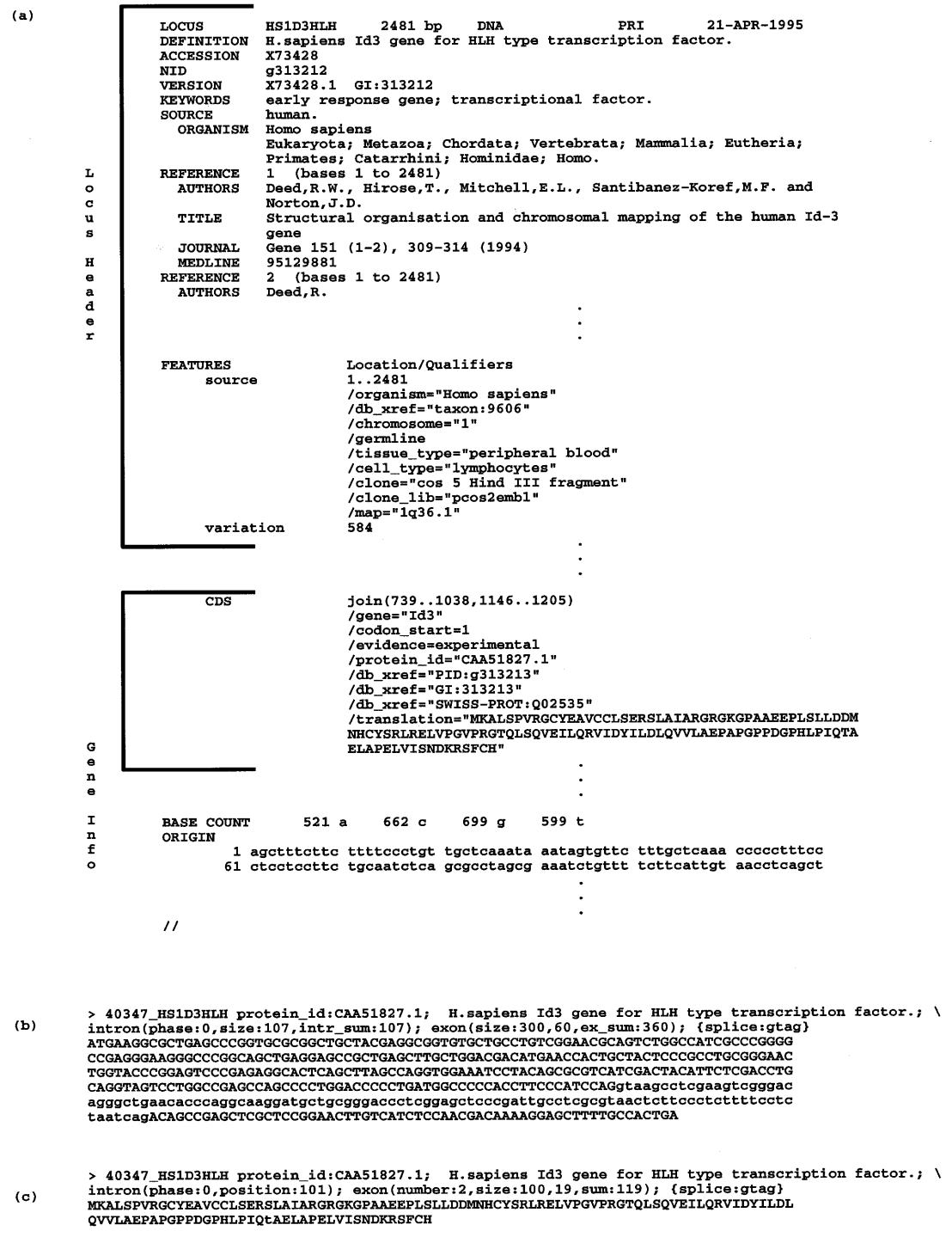

Figure 1. (**a**) Sample GenBank entry that carries an intron-containing gene. Sections marked as 'Locus Header' and 'Gene Info' are placed in the header file of the EID. They provide the information common to the entire locus and the individual gene, respectively. (**b** and **c**) The corresponding entries in the DNA and protein subsets of the EID, respectively. The "\" symbols are not present in the actual entries but are shown in the figure to indicate line continuation.

The database contains a total of 51 289 genes. To obtain a measure of redundancy in the database we purged the EID using the program GBPURGE (18) at 99% protein identity level. This reduced the number of genes to 42 460, implying that ~17% of the database is redundant. Since redundancies can be of many types, e.g. alternative splicing isoforms, close homologs, independently-sequenced genes, the purging, if needed at all, would have to be specific to the user's task. For that reason we choose to present the complete database, providing a tool for users to perform their own purges. To pull

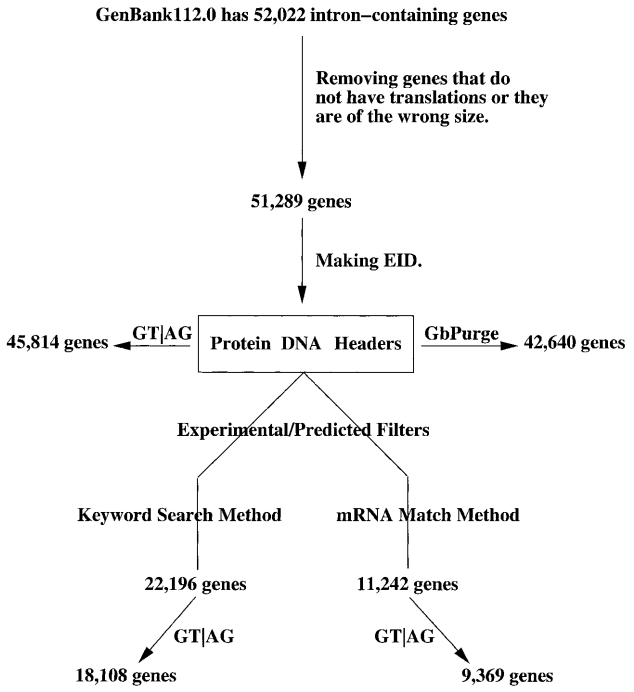

Figure 2. Flow diagram that briefly outlines the process of making the EID and filtering it as described in the paper. The arrows marked with 'GT|AG' represent the exclusion of all the genes that include non-'GT…AG' introns.

our genes with specific properties the EID can also be subjected to other filters, the application of which is outlined in Figure 2. We note that the use of the 'GT…AG' filter eliminates a gene if any one of its introns is not of the 'GT…AG' type. For that reason, while the proportion of non-canonical or possibly non-canonical introns in the EID is 3.6%, the proportion of eliminated genes is several times higher at 10.7%, reflecting the fact that most genes contain several introns.

DETAILS OF THE PREDICTED/EXPERIMENTAL FILTERS

The most important filter presented here works to eliminate those genes whose intron–exon structures have not been determined experimentally. We must note that because GenBank does not enforce the flagging of such sequences by the submitters, it may not be possible to determine with perfect certainty whether a particular gene is experimentally or non-experimentally determined.

We pursued two approaches to isolating those genes whose exon–intron structures were obtained experimentally. Under the first method, we parsed the GenBank.seq files mentioned above to extract all the mRNA-containing entries, which can be identified by 'mRNA' tags in LOCUS lines. We used BLAST2 (19) to search for matches between the coding sequences of the genes in the EID and the sequences in the mRNA database. By using a 95% identity cutoff, we isolated 11 242 genes whose structures were reliable. The results were placed in the following files: gb112.exp_mrna.pEID, gb112.exp_mrna.dEID and gb112.exp_mrna.hEID.

The second approach was based on a set of keywords that were applied to different regions within the GenBank entries. We believe that this method allowed us to filter out the vast majority of proteins with predicted gene structures. For the sake of flexibility, we split the filtering procedure into two steps. The first filter used two sets of keywords. The first set consisted of the names of various gene finders (Table 1a). If the filtering program found one of the names in the part of the GenBank entry common to all the features in the locus, all the genes in the locus were discarded. If it encountered one of the keywords in the part that was specific to a particular gene, only that gene was discarded. The other set of keywords consisted of 'evidence=experimental', 'non_experiment' and 'predict'. The program looked for these strings only in those portions of GenBank entries that were specific to individual genes (Fig. 1). It thus eliminated all the genes that were flagged by keywords 'non_experiment' and 'predict' while retaining all the genes that were labeled with the 'evidence=experimental' tag. The 'evidence=experimental' tag was stronger than all the other indicators and thus we kept all the genes that were flagged with it, regardless of which other flags may have been present.

Table 1. Frequencies of genes marked by the tags used in the keyword-based filtering of the EID

(a)

Tag	Number of Genes	Tag	Number of Genes
GENEFINDER	22,921	Other Gene Finders	27
NETPLANTGENE	4,283	"not_experimental"	8,097
GRAIL	4,203	"predict"	2,914
GENSCAN	4,018	"evidence=experiment"	545
HEXON	109		

(b)

Tag	Number of Genes	Tag	Number of Genes
BAC	2,036	HTG	678
PAC	140	"chromosome" {name}	364
YAC	0	"evidence=experiment"	545
cosmid	420		

(**a**) The tags from the first stage of the filtering procedure. All the counts represent the numbers of affected genes in the EID. The left column lists the five most frequent gene finding programs. The total count of all the remaining gene finders is presented under 'Other Gene Finders'. For an extensive and up-to-date list of gene finding programs see Dr Wentian Li's web page at http://linkage . rockefeller.edu/wli/gene/programs.html . (**b**) The tags from the second stage of the filtering process. All the counts represent the numbers of affected genes in the EID after the first filtering stage. Chromosome names were identified as words that either started with digits or consisted entirely of roman numerals.

The second filter attempted to remove all the remaining genes that had been found through large-scale genome sequencing projects. It has been our experience that most of such genes have not been confirmed experimentally. Thus, the second filter removed all the loci that contained words 'BAC' (Bacterial Artificial Chromosome), 'YAC' (Yeast Artificial Chromosome), 'PAC' (P1-derived Artificial Chromosome) or 'cosmid' in the DEFINITION line. It also removed those loci that contained the word 'chromosome' followed by the chromosome name. Finally, the filter discarded all the loci that had 'HTG' (High-Throughput Genomic sequences) as one of the tags in the keyword field. It should be noted that, as was the case with the first filter, any gene that contained the 'evidence=experimental' tag was retained regardless of the presence of other flags. The entries were spot-checked and examined with the corresponding literature to verify that the intron positions were not

Table 2. Frequencies of genes from the five most common species and all the species combined

	EID			After Keyword-based Filter			After mRNA-matching Filter		
Species	**total**	GT…AG	unknown	**total**	GT…AG	unknown	**total**	GT…AG	unknown
C. elegans	**17,138**	16,231	47	**316**	278	4	**453**	399	4
A. thaliana	**9041**	8,581	16	**940**	788	13	**860**	762	6
H. sapiens	**6355**	5,110	712	**4,850**	3,701	712	**4,274**	3,425	574
M. musculus	**2292**	1,847	183	**2,244**	1,804	183	**1,466**	1,217	100
S. pombe	**1900**	1867	2	**274**	252	2	**144**	140	1
All species	**50,697**	45,776	1,505	**21,605**	18,071	1,456	**11,220**	9,365	907

The 'GT…AG' columns refer to genes with only 'GT…AG' motif introns. A gene is labeled 'unknown', if it is unknown whether all the introns are of the 'GT…AG' type. Only nuclear-encoded genes are counted.

Table 3. Frequencies of most common intron splice motifs in the five most common species and all the species combined

All species	Entire EID	After keyword-based filter	After mRNA-matching filter	**H. sapiens**	Entire EID	After keyword-based filter	After mRNA-matching filter
gt…ag	227,185	76,202	47,906	gt…ag	33,502	23,410	20,772
NN…NN	2,753	2,737	1,701	NN…NN	1,224	1,224	968
gc…ag	949	495	274	gc…ag	166	126	99
gt…NN	261	243	118	gt…NN	110	110	66
NN…ag	184	165	89	NN…ag	95	95	59
gg…ca	182	175	84	gg…ca	34	31	21
EEEE	159	82	15	gt…ca	33	30	18
ta…gg	121	107	63	gt…gg	27	21	9
gg…ta	111	107	59	tg…ag	24	22	8
NN…tc	101	0	4	ta…gg	22	12	17
All Motifs	234,767	82,247	51,204	All Motifs	35,666	25,419	22,251

M. musculus	Entire EID	After keyword-based filter	After mRNA-matching filter	**C. elegans**	Entire EID	After keyword-based filter	After mRNA-matching filter
gt…ag	9,625	9,199	6,122	gt…ag	94,511	1,585	2,713
NN…NN	451	451	304	gc…ag	154	9	9
gc…ag	46	44	27	NN…tc	97	4	0
gt…NN	38	38	23	ga…NN	95	8	0
NN…ag	26	26	14	aa…NN	33	2	0
gt…ta	14	14	8	NN…tt	32	0	0
gg…ag	14	14	11	NN…aa	30	0	0
gg…ca	14	14	9	NN…cc	29	2	0
ta…gg	11	11	9	NN…ga	27	0	0
gt…ca	11	11	4	NN…tg	25	4	0
All Motifs	10,465	10,035	6,650	All Motifs	95,479	1,683	2,743

A. thaliana	Entire EID	After keyword-based filter	After mRNA-matching filter	**S. pombe**	Entire EID	After keyword-based filter	After mRNA-matching filter
gt…ag	46,012	4,564	5,067	gt…ag	3,976	626	290
gc…ag	282	37	45	gc…ag	7	2	1
NN…NN	52	49	14	EEEE	5	0	0
gg…ca	25	25	12	NN…NN	5	5	2
gt…ac	18	7	2	ta…ag	3	2	1
ta…gg	17	17	10	ta…gg	2	2	0
EEEE	16	3	2	ta…ga	2	2	0
ga…ag	9	6	3	tg…gg	2	2	0
gg…ta	9	9	5	ga…tc	1	1	0
gt…ga	8	5	0	cc…gt	1	1	0
All Motifs	46,637	4,846	5,216	All Motifs	4,022	660	296

Only nuclear-encoded genes are counted. 'EEEE' stands for introns less than four nucleotides long; and 'NN' represents unknown motifs.

computer-predicted. After the application of the two filters, the database was left with 22 345 entries which were placed in files with prefix 'gb112.exp_keyw2'.

STATISTICS OF THE DATABASE

The database clearly contains a wealth of interesting biological information. Here, we present some statistics that are relevant to the database construction and to some of the filters discussed earlier. Table 1 shows the incidence of the keywords used in the keyword-based filtering approaches. We note the frequent occurrence of the keyword GENEFINDER in the database. This is due to the fact that the GeneFinder program was used to locate genes in the sequence data produced by the *Caenorhabditis elegans* genome project. Indeed, as Table 2 shows, the *C.elegans* genes constitute fully a third of the entire database.

More generally, Table 2 lists the frequencies of genes from the top five species and all the species combined in the EID and its two filtered subsets. In presenting these statistics, we discard genes that, based on GenBank annotations, originate in organelles. One notable feature of the table is the sharp decrease in the number or *C.elegans* and *A.thaliana* genes, when predicted genes are filtered out. This is not particularly surprising since both the *C.elegans* and *A.thaliana* projects have focused on sequencing the genomic DNA, and not the mRNA. In the same vein, one can explain why the number of human genes stays relatively constant—many more human mRNAs have been sequenced. Table 3 presents the frequencies of the most common splice motifs for the same five species and for all the species combined. As above, we focus on nuclear introns. We note that the 'GT…AG' motif dominates the database accounting for 98% of all the known motifs. This percentage drops somewhat in the filtered subsets because the entries for experimentally-determined genes are intrinsically more susceptible to typographical errors. Indeed, one can easily force a computer program to predict introns with canonical boundaries only, while experimentally found introns suffer from occasional mistakes made by human annotators.

TOOLS/DISTRIBUTION

To facilitate interaction with the database, we provide a series of command-line tools, all of which can also be accessed through a menu-driven PERL program. One of the tools we provide lets the user purge the EID or its subset at 99% identity. Since we had already generated lists of all related proteins using the 'smallfamily' option of the program GBPURGE (18), this purging process is swift and flexible. We also offer a number of scripts to generate various databases—databases of individual exons, individual introns, coding sequences—all in FASTA format. Additional tools allow the users to restrict their analyses to various subsets of the database. One can isolate genes (or exons, or introns) by species, or by splicing motifs. We provide a script to isolate nuclear- and organelle-encoded genes. It must be kept in mind, however, that the script is based on the 'ORGANISM' field within GenBank, and is thus not perfectly effective. The user can also create different databases for introns of different phases or exons with different flanking phases. More generally, we provide a script that uses a PERL regular expression as one of the command line parameters to extract all the database entries that contain matches for that regular expression. One can thus create custom databases based on function or other gene features. In addition, we include utilities that allow the user to look up entries by EID indices, protein_id codes, LOCUS names or ACCESSION codes. All these tools, as well as the EID and its filtered subsets are provided within a single distribution. The distribution also includes the parsers that were used to make the database, as well as the keyword filter that was used to exclude genes whose structures have not been found through experiment. The distribution can be found at http://mcb.harvard.edu/gilbert/EID

ACKNOWLEDGEMENT

We thank Dr Carl Rosenberg for making his program available to us.

REFERENCES

1. Gilbert,W. (1978) *Nature*, **271**, 501.
2. Doolittle,W.F. (1978) *Nature*, **272**, 581–582.
3. Gilbert,W. (1987) *Cold Spring Harbor Symp. Quant. Biol.*, **52**, 901–905.
4. Long,M., Rosenberg,C. and Gilbert W. (1995) *Proc. Natl Acad. Sci. USA*, **92**, 12495–12499.
5. Tomita,M., Shimizu,N. and Brutlag,D.L. (1996) *Mol. Biol. Evol.*, **13**, 1219–1223.
6. Logsdon,J.M.,Jr (1998) *Curr. Opin. Genet. Dev.*, **8**, 637–648.
7. Logsdon,J.M.,Jr, Stoltzfus,A. and Doolittle,W.F. (1998) *Curr. Biol.*, **8**, R560–R563.
8. Bagavathi,S. and Malathi,R. (1996) *FEBS Lett.*, **392**, 63–65.
9. Long,M., de Souza,S.J., Rosenberg,C. and Gilbert,W. (1998) *Proc. Natl Acad. Sci. USA*, **95**, 219–223.
10. Cho,G. and Doolittle,R.F. (1997) *J. Mol. Evol.*, **44**, 573–584.
11. Rzhetsky,A., Ayala,F.J., Hsu,L.C., Chang,C. and Yoshida,A. (1997) *Proc. Natl Acad. Sci. USA*, **94**, 6820–6825.
12. de Souza,S.J., Long,M., Klein,R.J., Roy, S, Lin,S. and Gilbert,W. (1998) *Proc. Natl Acad. Sci. USA*, **95**, 5094–5099.
13. Deutsch,M. and Long,M. (1999) *Nucleic Acids Res.*, **27**, 3219–3228.
14. Benson,D.A., Boguski,M.S., Lipman,D.J., Ostell,J., Ouellette,B.F., Rapp,B.A. and Wheeler,D.L. (1999) *Nucleic Acids Res.*, **27**, 12–17. Updated article in this issue: *Nucleic Acids Res*. (2000), **28**, 15–18.
15. Stoltzfus,A., Logsdon,J.M.,Jr, Palmer,J.D. and Doolittle,W.F. (1997) *Proc. Natl Acad. Sci. USA*, **94**, 10739–10744.
16. Hall,S.L. and Padgett,R.A. (1996) *Science*, **271**, 1716–1718.
17. Brown,J.W., Smith,P. and Simpson,G.G. (1996) *Plant Mol. Biol.*, **32**, 531–535.
18. Rosenberg,C. The program GBPurge can be obtained from http://www.fallingrain.com/
19. Altschul,S.F., Madden,T.L., Schäffer,A.A., Zhang,J., Zhang,Z., Miller,W. and Lipman,D.J. (1997) *Nucleic Acids Res.*, **25**, 3389–3402.