12. The Processing of Biological Sequence Data at NCBI

by Karl Sirotkin, Tatiana Tatusova, Eugene Yaschenko, and Mark Cavanaugh

Summary

The biological sequence information that makes up the foundation of NCBI's databases and curated resources comes from many sources. How are these data managed and processed once they reach NCBI? This chapter will discuss the flow of sequence data, from the management of data submission to the making of data available for use.

Overview

The sequence database GenBank (Chapter 1) is made up of nucleotide sequences submitted by individual scientists and sequencing centers from around the world. These sequences have been submitted directly to GenBank or are replicated from one of the collaborating databases,the European Molecular Biology Laboratory (EMBL) Data Library or the DNA Data Bank of Japan (DDBJ). Once processed, the data are made available for public use in several ways (Table 1). The Entrez search and retrieval system (Chapter 14) can be used to find one or many sequences in GenBank by using text queries into fields that are linked to the raw sequence data, such as author, definition line, organism, and so on. The sequences themselves are searched directly by using BLAST (Chapter 15), which takes a sequence as a query to find similar sequences. Large subsets or the latest, complete version of GenBank can also be downloaded by FTP. Underlying the submission, storage, and access processes seen by users of GenBank, BLAST, and other curated data resources [such as the Reference Sequences (Chapter 17), the Map Viewer (Chapter 19), or LocusLink (Chapter 18)] is an information management system that consists of two major components, the ID database and the IQ database. Whereas ID handles incoming sequences and feeds other databases with subsets to suit different needs, IQ holds links between sequences from ID and links from these sequences to other resources. This chapter discusses these resources in more detail.

Table 1. Access to genetic sequences at NCBI.

Access	URL
FTP	http://www.ncbi.nlm.nih.gov/Sitemap/index.html#FTPSite
BLAST	http://www.ncbi.nlm.nih.gov/BLAST/
Entrez	http://www.ncbi.nlm.nih.gov/Sitemap/index.html#Entrez

Abstract Syntax Notation 1 (ASN.1) Is the Data Format Used by the ID System

ASN.1 is the data description language in which all sequence data at NCBI are structured. ASN.1 allows a detailed description of both the sequences and the information associated with them, e.g., author names, source organism, and sequence features. Maintaining all of the data in the same structured format simplifies data parsing, manipulation, and quality assurance, as well as making data integration and the development of software for sequence analysis easier. All of the various divisions of GenBank can be downloaded in ASN.1 from the NCBI FTP site. Similar to an XML DTD, ASN.1 has associated with it a description of the legal data structures; this file is called asn.all and is available by FTP in the ASN directory from the file, ncbi.tar.Z. The DEMO directory of this same file contains the tool, testval, which validates the data against asn.all. There is also a set of utilities for producing ASN.1 while programming in C; these are largely in the subutil.c file of the API directory. In the ID data management system, data are stored as ASN.1 blob, minimizing the amount of biological information that needs be captured and updated in the relational database schema.

Sources of Sequence Data

The sequence data available at NCBI comes from many different sources (Figure 1). In summary, the data consist of:

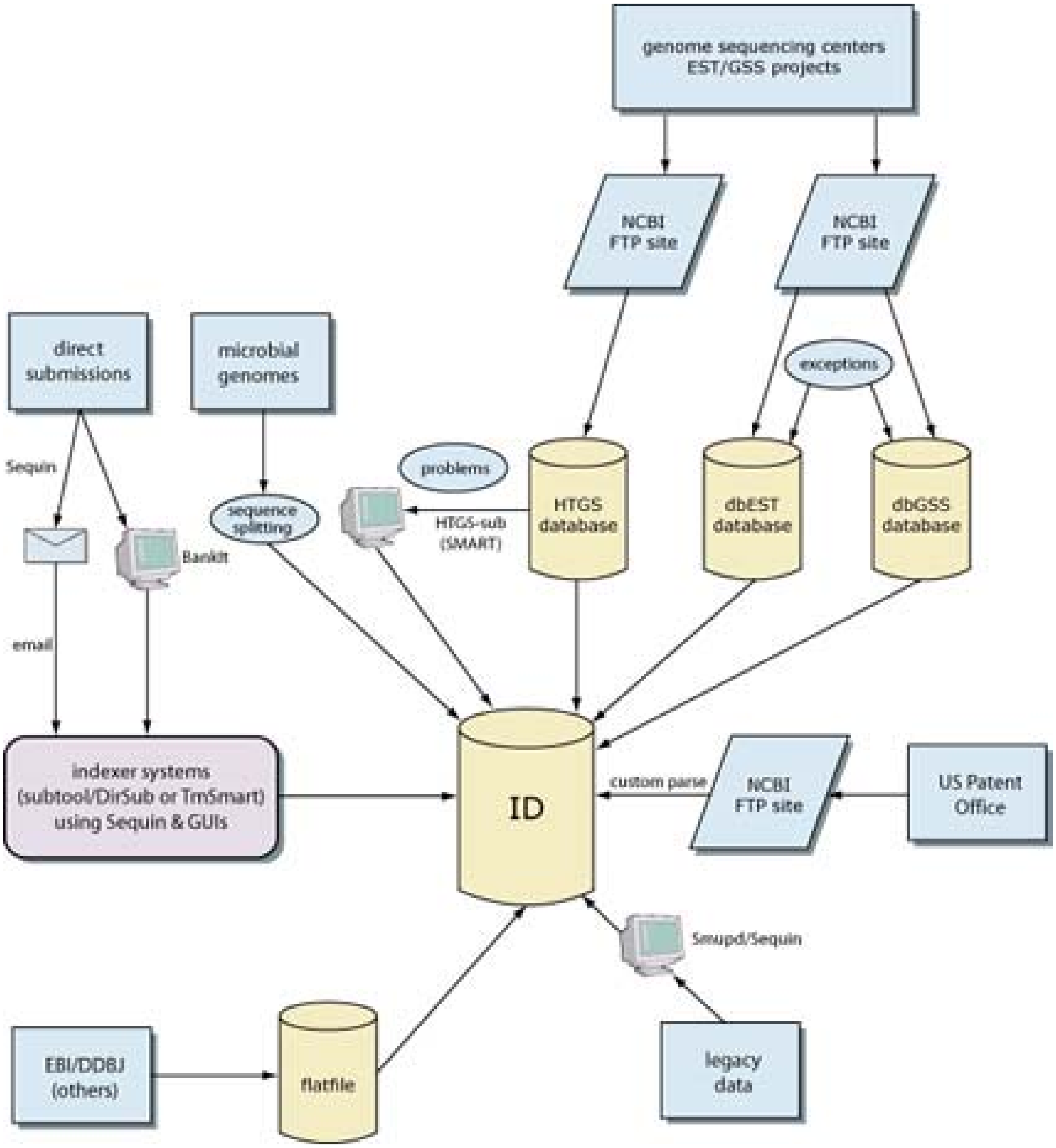

Figure 1: Sources of sequence data available at NCBI.

1. GenBank sequences (Chapter 1)
2. Reference sequences (Chapter 17)
3. Sequences from other databases, such as SWISS-PROT, PIR, PRF, and PDB
4. Sequences from United States patents

The submission pathway depends on the data source (see Figure 1) and volume. Generally, large-volume submitters, such as HTGS, use FTP, often after using tools such as fa2htgs to convert their data to ASN.1. Small-volume submitters typically use either BankIt (Chapter 1) or Sequin (Chapter 11) to prepare the ASN.1 for submission.

The data received are subject to some quality control. The submission tools BankIt, Sequin, and fa2htgs have built-in validation mechanisms that ensure that the data submitted have the correct structure and contain the essential information. The work of the GenBank indexing staff, who also use Sequin, adds an additional layer of quality control and provides assistance with problems or complex submissions and updates.

Data Flow Components

The ID Database

The ID database is a group of standard relational databases that holds both ASN.1 objects and sequence identifier-related information. ASN.1 objects follow the specifications in the asn.all file for NCBI sequence data objects. ID holds data for GenBank and other data, all of which are included in Entrez. All of the sequences for the International Nucleic Acid Database Collaboration are in GenBank and must have Accession numbers assigned to them. These Accession numbers define an entity, the sequence of which is described. When the understanding of that sequence changes, the sequence can have a new version. This leads to two parallel ways of tracking sequence versions for an object. In the GenBank flatfile format, there is an Accession.Version, where the version gives the ordinal instance (version) of the sequence. Within ID, each unique sequence is assigned a gi number; therefore, the chain of gi numbers for an Accession also gives all of the instances of the sequence for that Accession. The chain identifier within ID can be used to link these gi numbers. (Not all sequences within ID are in GenBank and not all have sequence versions, but all sequences have a chain of gi numbers. For this reason, internally, the gi number is the universal pointer to a particular sequence, as opposed to the Accession.Version, which would work only for versioned sequences.) The ID database is also the controller for allowed “takeovers” of one Accession by another. A takeover would occur, for example, when two clones' sequences were now to be merged into a single clone. One or both of the two older clones' Accessions could be taken over by the new clone. The actual details of the architecture of this group of databases and software associated with it are described later in this chapter.

Output of Data from the ID System

Once all incoming data have been converted to ASN.1 format and have been entered into ID, the data are then replicated to several different servers and also transformed into several different formats (Figure 2). The replication is necessary for a number of reasons: (1) it separates the “incoming” data system (ID) from the “outgoing” data, i.e., the data served in response to scientific queries by users; (2) replicating the data to different servers helps balance the load of queries, thus providing quicker response times, and allows different servers to specialize in different functions; and (3) it protects against data loss; having copies kept at different locations means that the data are safe, should one server fail.

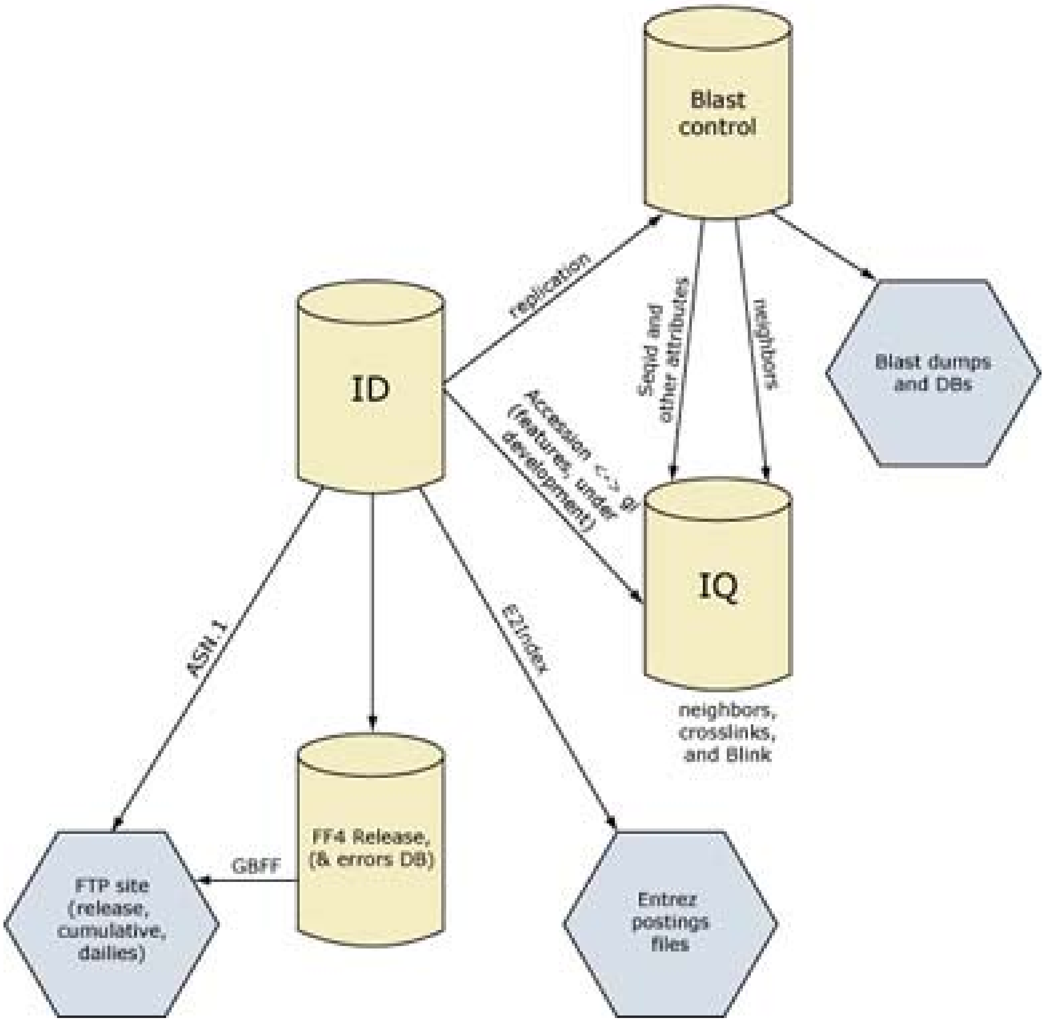
Blast control
replication
Seqid and other attributes
neighbors
Blast dumps and DBs
ID
Accession <-> gi (features, under development)
IQ
neighbors, crosslinks, and Blink
ASN.1
E2Index
FF4 Release, (& errors DB)
GBFF
FTP site (release, cumulative, dailies)
Entrez postings files

Figure 2: Products of the ID system.

The IQ Database

The IQ database is a Sybase data-warehousing product that preserves its SQL language interface but inverts the data it stores so that the data are stored by column, not by row. Its strength is that it can speed up results from queries based on the anticipated indexing. This non-relational database holds links between many different objects.

For example, as part of the processing of incoming sequences, each protein and nucleotide sequence is searched (using BLAST, Chapter 15) against the rest of the database for similar sequences. The resulting set of similar sequences (sometimes known as “neighbors”) are made available to users by selecting the **related sequences** link next to a record in Entrez (Chapter 14). The IQ database keeps track of the neighbors for any given sequence; these relationships are all precomputed to save time for the people using this feature.

As well as storing the relationships between nucleotide or between proteins, IQ also holds information on the links between entries in different Entrez databases. This might include, for example, information on the publications cited within sequence records (which link to PubMed), or to an organism in the Taxonomy database. Some of this information comes from the analysis of the ASN.1 in ID by e2index.

The BLAST Control Database

The BLAST Control database receives information from ID and uses it to generate BLAST databases (Chapter 15) for the BLAST query service and for stand-alone BLAST users, as well as for internal use, to generate the sequence neighbors detailed in IQ.

The GenBank Flatfile and Error Capture Databases

Many people who use NCBI services think of the GenBank flatfile as the archetypal sequence data format. However, in terms of the ID internal data flow system, ASN.1 is considered the original format from which reports such as the GenBank flatfile can be generated. Generally, the GenBank flatfile is generated on demand from the ASN.1. However, for certain products, such as complete GenBank releases, a GenBank flatfile image is made for each active sequence and is stored in a database called FF4Release, which consists of the latest transformation of ASN.1 to the GenBank flatfile format.

This database also doubles as the place where internal error reports are captured. The reports can be analyzed and displayed for different time points in the data processing pathway: (*a*) ASN.1 itself can be validated using the testval tool. (Syntax checking is not necessary, because the underlying ASN.1 libraries enforce proper syntax according to the definition file.); (*b*) errors can be discovered during conversion to GenBank flatfile format; and (*c*) through a reparse from GenBank flatfile format to ASN.1. This is done as a further check for legality of the ASN.1, and our current software for producing GenBank format reports from it.

Entrez Postings Files

When sequences are submitted to GenBank or one of the collaborating databases, useful information about the sequence is often also included. This might include a brief description of a gene in the definition line, along with annotated sequence features, e.g., the source organism name. To make this information searchable via Entrez, these words have to be indexed. They are therefore extracted from the ASN.1 (using a tool called e2index) and are then stored in the Entrez posting files, which are optimized for Boolean queries by the Entrez system (see Chapter 14).

All of these products from the ID system are listed in Table 2. NCBI also generates weekly "LiveLists" for public, collaborator, and in-house use. LiveLists show currently known Accession numbers that are in use (i.e., they have not been replaced or otherwise removed from circulation for one reason or another).

Table 2. Products of the ID system.

Type	Source	ASN.1	GBFF[a]	Qscore	GenPept	Protein FASTA
Cumulative	GenBank	X		X	X	X
Incremental	GenBank	X		X	X	X
Incremental	GenBank[b]		X	X		
Cumulative	RefSeq	X	X		X	X
Incremental	RefSeq	X	X		X	X

[a] GBFF, GenBank flatfile.

[b] NCBI records only.

Data Flow Architecture

ID has a distributed architecture, the features of which are shown in Figure 3. The system is activated when a client (internal to NCBI) is ready to load data into the ID system. A stand-alone program takes ASN.1-containing files, while a client API can submit data to ID through the IDProdOS. The rationale behind the various architecture components is discussed below.

IDProdOS is an open server (commonly called "middleware"), which sits between the clients and the database system. It hides details of the underlying complexity from the client API. For example, the previous version of the ID system consisted of a single database. "Using the Open Server" meant that when the conversion to the current system took place, only the open server had to be changed, and none of the clients. IDProdOS begins the complex process of checking that the actions implied by the load are allowed and changes the SeqId pointers to gi numbers to be used as sequence-specific pointers. For example, in a record that has DNA and protein sequences, including annotation and sequence identifiers, the identifier on the protein has to be unique. If the identifier has been used previously on another DNA sequence and the current sequence is not replacing the previous sequence, then this is not allowed because proteins are not allowed to move between GenBank records. As an exception, this rule is sometimes relaxed for proteins moving between segments of a complete genome submission.

The IdMain database contains the sequence identifiers for each of the sequence records, including all those for ASN.1 blobs that contain multiple sequences. It enforces sequence version rules, among others.

Relational satellite databases are fully normalized databases that hold records for which there is only one sequence per intended ASN.1 blob. Few, if any, features are allowed on records intended for relational satellite databases. (The PubSeqOS produces the ASN.1 by querying the relational tables.) This is in contrast to the Blob satellite databases, from which ASN.1 is retrieved pre-made.

Blob satellite databases, although relational databases, contain ASN.1 objects as unnormalized data objects.

The SnpAnnot database contains only feature information. It has simple mutation information from dbSNP (Chapter 5), which is added to NCBI-curated records by the PubSeqOS when the records are requested.

To visualize the role of replication, the rectangle in the middle of Figure 3 represents the use of the Sybase Replication Server to copy information from the loading side of the system to the query side.

PubSeqOS is an open server (commonly called "middleware") that sits between the clients and the database system. It hides details of the underlying complexity from the client API. It actually has an almost identical code base to IDProdOS because it serves similar functions. When the ASN.1 that has been requested is to be presented in a format other than ASN.1, psansconvert is called to do the conversion. This separate process allows both insulation from any possible instability and allows for use of multiple central processing units in a natural way.

The EntrezControl and The Graveyards databases are uniquely on the query side. The EntrezControl database controls information on records that await indexing and have been indexed by e2index.The Graveyards are blob databases and contain those blobs killed by replacement or that were taken over by other blobs. Once taken over, blobs are not changed; therefore, they need not be on the loading side.

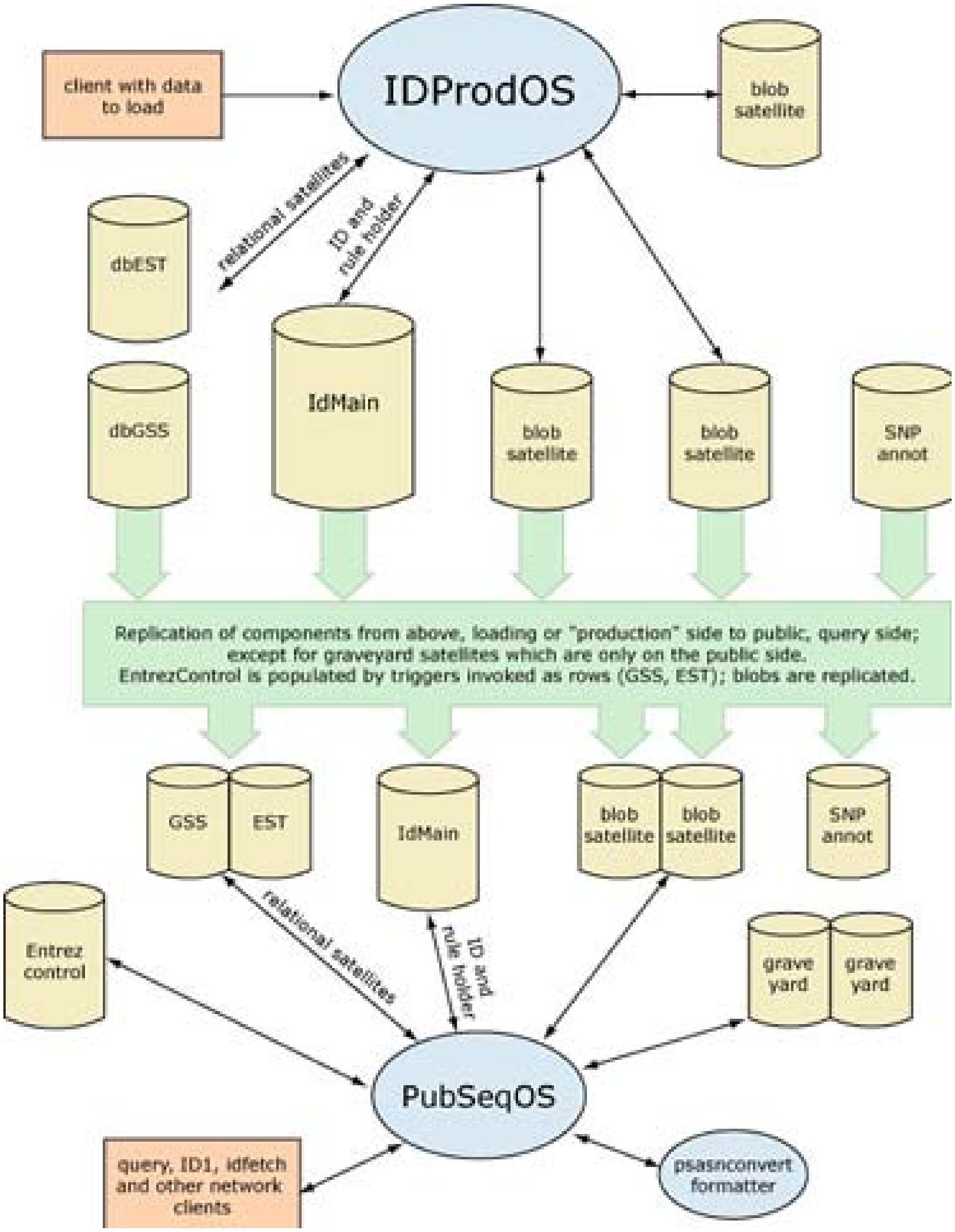
client with data to load
IDProdOS
blob satellite
relational satellites
ID and rule holder
dbEST
dbGSS
IdMain
blob satellite
blob satellite
SNP annot
Replication of components from above, loading or "production" side to public, query side; except for graveyard satellites which are only on the public side. EntrezControl is populated by triggers invoked as rows (GSS, EST); blobs are replicated.
GSS
EST
IdMain
blob satellite
blob satellite
SNP annot
Entrez control
relational satellites
ID and rule holder
grave yard
grave yard
PubSeqOS
query, ID1, idfetch and other network clients
psasnconvert formatter

Figure 3: The ID system architecture.