Nucleic Acids Research, 1996, Vol. 24, No. 1 57–63

Improvements to the GDB™ Human Genome Data Base

Kenneth H. Fasman*, Stanley I. Letovsky, Robert W. Cottingham and David T. Kingsbury

Division of Biomedical Information Sciences, Johns Hopkins University School of Medicine, Baltimore, MD 21205, USA

Received October 4, 1995; Accepted October 5, 1995

ABSTRACT

Version 6.0 of the Human Genome Data Base introduces a number of significant improvements over previous releases of GDB. The most important of these are revised data representations for genes and genomic maps and a new curatorial model for the database. GDB 6.0 is the first major genomic database to provide read/write access directly to the scientific community, including capabilities for third-party annotation. The revised database can represent all major categories of genetic and physical maps, along with the underlying order and distance information used to construct them. The improved representation permits more sophisticated map queries to be posed and supports the graphical display of maps. In addition, the new GDB has a richer model for gene information, better suited for supporting cross-references to databases describing gene function, structure, products, expression and associated phenotypes.

INTRODUCTION

The GDB Human Genome Data Base, an international collaboration hosted at Johns Hopkins University, is the public repository for human genomic mapping data and supporting information (1–4). Iterations of the database design from its initial release through version 5.6 can all be easily traced to GDB's predecessor, the Human Gene Mapping Library (HGML; ref. 5). With version 6.0, GDB has been completely redesigned to address some well-recognized shortcomings of the previous versions: (i) Text-based representations of map order and distance were cumbersome for capturing complex genome mapping data. No tool was available for the graphical display of GDB maps; (ii) It was sometimes difficult for the occasional user of GDB to find genes of interest in the database. Genes in GDB were only modestly linked to related information in other databases; (iii) Data acquisition and curation were funneled solely through HUGO editorial committees and GDB staff, making it difficult for the community to directly contribute to the database.

The focus of the development of version 6.0 of the Human Genome Data Base was the near-complete redesign of the database schema to address these inadequacies. We will present the most important features of the new GDB, as well as planned enhancements for the near future.

FEATURES OF VERSION 6.0

The new GDB design reflects a number of major changes in approach over previous versions of the database, including: a change in the modeling formalism used to design the database, from a relational model to a more object-oriented approach; a step towards allowing community-based curation; enhanced representations of maps and genes that can support a greater diversity of opinion in the community about features of the genome and their locations; graphical display and editing of genetic and physical maps and graphical user interfaces for all aspects of browsing, querying and editing the database.

Object-oriented data model

The Object Protocol Model (OPM) developed by Markowitz and colleagues at Lawrence Berkeley National Laboratory, is a representation for database schemas supported by a set of software tools (6). (These tools include object-to-relational translation facilities, so although GDB 6.0 will be specified in terms of objects, the underlying database management software will continue to be Sybase, a relational system). The OPM schema representation is essentially object-oriented; a schema consists primarily of a set of object classes, each of which has a set of attributes. For example, in a personnel database one might have a class *Employee* with attributes *name*, *supervisor*, *salary*, *projects* and so on. A database is then a collection of objects, each of which belongs to one or more classes. The attributes of a given object are determined by the classes it belongs to. If there is an *Employee* object representing Jane Doe, it can specify her name, boss, pay, work in progress, etc.

Attributes have associated datatypes, which can be primitive (e.g. numbers or character strings) or other classes defined in the schema, including object classes and controlled vocabulary classes (restricted sets of values defined in the database). Attributes can hold a single value or multiple values. Finally, attributes can be derived, that is, computed from other data by means of a formula. Derived attributes are not stored and thus cannot be directly edited; to edit a derived value one must edit the actual values from which it is computed.

Object classes are grouped into a hierarchy of sub- and super-class relationships, often called an *isa* hierarchy (as in dog *isa* mammal and mammal *isa* animal). A class is said to inherit all the attributes pertaining to its superclasses, their superclasses and so on. Thus each class has its own locally defined attributes, as

* To whom correspondence should be addressed

well as attributes inherited from all of its ancestors in the *isa* class hierarchy.

A database designed with OPM (and object-oriented models in general) is easier to develop and understand than an equivalent relational database. This is because the relationships between classes and their attributes and between pairs of classes, are more explicitly defined. OPM allowed the GDB staff to develop the 6.0 design more quickly than would have been possible using the relational model. More importantly, the new database schema is easier for the average user of GDB to comprehend and therefore easier to query.

New curatorial model

GDB 6.0 supports direct community curation, interactively or via bulk submissions. Consequently, it must address issues of data ownership and editing permissions. The starting point for our design is the principle that each item in the database will have an owner who has exclusive editing privileges on all the stored attributes of that object. This owner may be a single individual, or a group such as a laboratory that shares a single GDB account (certain standard objects such as chromosomes and cytogenetic bands will belong to GDB itself). This principle provides a degree of accountability and control over the contents of the database without incurring the overhead of an attribute-level permission scheme.

However, it does not provide any means for community members to edit attributes of objects belonging to others, other than to request that the change be made by the owner. A loophole is provided by derived attributes, which are computed by a formula from other data and thus are not really part of the object insofar as editing rights are concerned. We use this to support third-party annotation of objects, since annotations appear only in the derived attributes *annotations*, *citations*, *externalLinks* and *aliases*. Annotations are like Post-It™ notes; they contain some text and they can be linked to one or more objects. Citations associate an object with a literature reference. External links can associate GDB objects with data in other databases or on the World Wide Web (WWW). Aliases supply alternative names for an object.

Each of these attributes has a corresponding class, which means that someone making an annotation on a *Gene* actually creates a new *Annotation* object, linked to the *Gene*, on which they have exclusive editing rights. A reference to that annotation object will automatically appear on the *annotations* attribute of the *Gene*, even if the *Gene* is owned by someone else, because the derivation formula for the *Gene*'s *annotations* attribute says find all annotations associated with this object.

Thus community members can add comments, literature references, arbitrary database links or aliases to any GDB object. Other third-party annotation attributes are defined lower in the *isa* hierarchy to allow the creation of links between objects in GDB and entries in other databases. For example, one can create links from *Variations*, *Probes* or *GenomeRegions* to entries in sequence databases such as GSDB or GenBank, or from *GenomeRegions* to phenotypes in OMIM.

As before, contributors can specify that the data they enter be held in confidence for up to 6 months after submission. During that period the data are in the database, but only the owner (and GDB staff) can retrieve or edit them. This allows GDB accession numbers to be assigned to the data in a manuscript prior to its publication.

There are a few exceptions to the rule about editing rights being vested in an object's owner. HUGO committee members (and GDB staff) will have the right to delete any object from the database. This is less drastic than it sounds, since the information about the object remains in an archive and the deletion can be reversed subsequently. All changes to the database are recorded and the change history of any object can be reconstructed upon request.

Anyone with a GDB account can add a map to the database. The database may hold different, inconsistent maps of the same region, submitted by different (or the same) researchers. Some maps will be marked as 'HUGO Approved' and will be maintained by a chromosome committee. Also, in a few cases we allow editing of one object to automatically cause changes in another object that may be owned by someone else, where the owner of the second object has explicitly permitted such a change. This mechanism can be used to add markers to certain maps, for example. Finally, only GDB staff can split or merge objects (e.g. when a duplication is detected in the database).

Improved map representation and querying

Maps are the central concern of the Genome Data Base. The GDB 6.0 representation of map-related information is designed to satisfy a number of goals:

Expressiveness. Maps must represent information about order and distance. We model a map as a list of markers in which each marker is assigned both a coordinate and a pair of flanking markers. The flanking markers provide order information which can be used to determine the precision of the coordinate assignment. Order-only maps can be accommodated using arbitrary, ordinal coordinates. A typical map is a combination of fully ordered 'framework' markers and other markers placed within specified framework intervals.

Flexible resolution. Owners of maps should be able to decide whether they want a particular *GenomeRegion* to be a point or an interval, that is, they should be free to choose the region's level of resolution. There is no attribute of *GenomeRegions* that specifies whether a given region is a point or an interval. Instead, map-related classes that use *GenomeRegions* as markers always refer to them by a pair of values: a *GenomeRegion* together with an 'endpoint specifier'. The latter is a controlled vocabulary attribute that can take on the values 'Start', 'End' or 'Entire'. If a marker is listed as 'CFTR Entire' then the map is treating the CFTR gene as a point. If it said, by contrast, 'CFTR Start' this would refer to the beginning of the CFTR gene. Note that the meaning of 'Start' and 'End' are relative to the orientation of the map as a whole. One map's start for a given region may be another map's end for that same region if the map orientations are reversed with respect to one another.

Graphic representation. Our map model lends itself readily to diagram generation. The natural style of diagram for this representation would display markers as points or intervals, depending on the endpoint specifiers used in the map, plus (optional) ambiguity bars based on the flanking marker information. The GDB map drawing program (see below) can extract these map representations from the database and draw, navigate and print them.

Explicit representation of spatial facts underlying maps. All spatial information will be searched and presented via maps, which will in turn be (optionally) annotated and augmented by *MapRelations*. The data that define a map, coordinates and flanking markers, are fundamentally provisional; is based on the best information available at the time the map was made. Persons interested in the map may want to know the underlying information, in order to assess the precision and accuracy of the map. We provide a representation for the underlying facts in terms of *MapRelations*, which come in two types: *TwoPointDistances* and *ThreePointOrders*. A person submitting a map is not required to create the corresponding relations, but they are free (and even encouraged) to do so.

Map relations can be useful for expressing knowledge about spatial arrangements that may be too fragmentary to construct a map. The process of converting map relation information into maps typically involves making decisions about which in a set of potentially conflicting relations to believe and which to throw out, how to compromise among different distance measurements and how to arbitrarily assign coordinates to poorly localized points. Some work has been done on automating aspects of this map assembly process (e.g. ref. 7). The GDB 6.0 schema is designed to support and encourage attempts along these lines, by allowing both maps and map relations to be stored. This will allow interested researchers to analyze *MapRelations* stored in the database and submit the maps they assemble from them.

Explicit representation of experimental data underlying maps. Experimental data underlying maps can be represented in terms of *ReagentRelations*, which describe relationships between mapping reagents. Subclasses of *ReagentRelation* include *AmplifiesFrom* (e.g. an STS-clone hit), *HybridizesWith* (e.g. a FISH probe) and *DerivedFrom* (e.g. subcloning). Reagent relations can be linked to the map relations they support. For example, an *AmplifiesFrom* might be linked to a *ThreePointOrder*, indicating that an STS can be said to fall within a clone's endpoints. Reagent relations can also be grouped into *MappingExperiments*, which can in turn be linked to maps.

Order and distance querying. The representation of maps must lend itself to efficient searching in a database. Our representation has been designed to support the following classes of queries: Query by position, find all maps or genome regions that overlap, contain or are contained in a specified genome region. The region can be specified as a single marker, a marker plus or minus a distance or as a range defined by a pair of markers; Query by order, find all maps [in]consistent with a specified marker order; Query by distance, find all maps [in]consistent with a specified inter-marker distance range.

The ability to efficiently perform queries of this sort, not possible previously in GDB, will provide the basis for automated map comparison algorithms in the future.

Community contribution to maps. The owner of a map can allow other users to add data directly to the map. This feature is a marriage of the concepts of third-party annotation and map representation to yield maps which can be enhanced by others. Maps are added to by creating *MapRelations* and linking them to the map. If the *MapRelation* can be used to place a new locus on the map and the map owner has enabled public editing, the locus will be added, with an automatically computed coordinate and flanking markers. The assignment of markers and coordinates is by a modest algorithm that should not be highly trusted. This mechanism will not replace intelligent map construction, but it will provide an interim view of the data between chromosome workshops. A marker placed on a map in this way will immediately be visible to persons downloading the map from the database. We will also make available map assembly algorithms that can do a more careful determination of coordinates and flanking markers from the underlying map relations and can be run on a periodic basis if the owner of the map so chooses. Map owners can also reject submitted relations and manually edit the coordinate and flanking marker assignments if they wish.

Support for large maps. As physical maps of the human chromosomes continue to develop, they are rapidly encompassing thousands, even tens of thousands, of markers and clones. Maps with this many objects will be difficult to render. However, the new map representation permits the expedient extraction of portions of larger maps by cleaving the map at framework marker positions. Map viewing software (see below) and *ad hoc* queries can be used to obtain a particular section of a map by specifying the markers flanking the region of interest. Note that the mechanism for community contribution to maps discussed above allows for the efficient addition of incremental changes to large maps as well.

Improved gene representation

The representation of genes in previous versions of the Human Genome Data Base was modest. It included an official HUGO symbol and name, alternative symbols, a consensus cytogenetic localization and method of assignment to that position. Gene records also included links to other databases for information on phenotype, nucleotide sequence, homology and enzymatic activity of the associated product.

The name attribute in particular was variously used to store information about fully or partially characterized protein products (e.g. secondary structure, size, location in 2-D gels), inferred function, associated phenotype, expression pattern, transcription/translation status (e.g. pseudogenes), method of identification, chromosome structure (e.g. fragile sites), sequence motifs (e.g. homeoboxes), homologies, viral integration sites and so on. It was difficult to search for genes in GDB based on this information, because it was not stored in a structured fashion using specific attributes and controlled vocabularies.

In the new GDB, there are object classes to represent genes, gene elements (e.g. exons, introns), regulatory regions, gene families (named collections of genes) and gene products (RNA or protein). The *Gene* class has attributes for linking component elements, products and regulatory regions and describing transcription/translation status and method of identification. These are not attempts to capture all of human biology in the Genome Data Base, but rather to serve as structured placeholders for searching and anchoring links to other genomic databases. The introduction of these classes and GDB 6.0's ability to manage *ad hoc* links to external databases provide a more robust method for integrating map information with biological data that are curated elsewhere.

Graphical map viewer

A program for the graphical display of maps accompanies the new database release. The GDB Map Viewer displays genetic and physical maps of the human genome using the common graphical conventions seen in the literature. The initial version displays

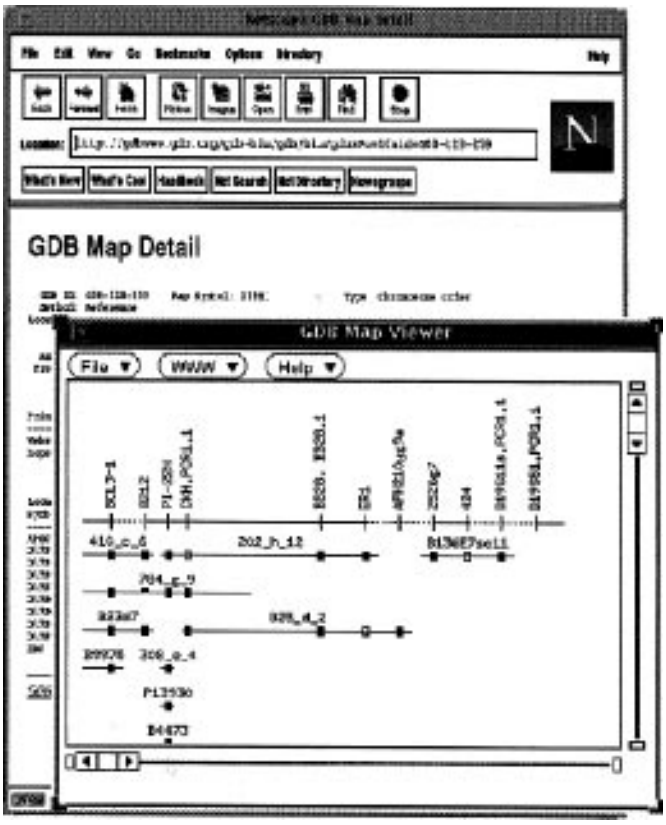

Figure 1. Map viewer displaying a physical map retrieved from the GDB Web server. Web clients (e.g., Netscape, Mosaic) can be configured to invoke the viewer automatically when a map is downloaded from GDB.

linkage, radiation hybrid, contig and cytogenetic maps. The program provides all the typical functions of graphical interfaces, including scrolling and zooming to browse large or dense maps, switching the orientation of the map and a basic printing capability.

The Map Viewer is designed to be used alone with locally saved GDB map files, or more commonly as an external viewer program in conjunction with World Wide Web browsers such as Netscape or Mosaic (Fig. 1). One can configure one's favorite browser so that the map viewer is automatically called when a map's hyperlink is selected from a GDB Web page. The viewer was written using software that allows one program to be developed simultaneously for many computer platforms. The viewer is initially available for the Macintosh, Microsoft Windows and Sun operating systems (other systems will be supported as demand warrants).

When operating in conjunction with a Web browser, the GDB Map Viewer has another important feature. Objects in the displayed map (e.g. markers, clones) can be selected with the mouse. The viewer will then communicate back to the GDB server via the Web browser to retrieve the details of the selected objects. This feature is initially available only with the Netscape browser.

Subsequent releases of the Map Viewer will support additional types of maps (e.g. gene structure and restriction maps) and multiple map alignments. They will provide more sophisticated printing functions, including the printing of selected regions of a map and printing complex maps across an arbitrarily specified print area (using multiple overlapping pages as required). The Map Viewer will also be able to save the displayed map(s) so that they may be exported to drawing programs or documents.

WWW interface

Initially, all interactive access to the GDB 6.0 database will be via a WWW interface. This interface is based on the Genera software (8) and provides browsing, querying, keyword searching and editing functions. It also provides an entry point for the Map Viewer, which is started up by a Web browser as an external viewer to display genomic maps.

Genera is a software tool set that simplifies the integration of Sybase- and OPM-based databases into the WWW. It can be used to retrofit a Web interface to an existing database or to create a new one. To use Genera one writes a specification of the database and of the desired appearance of its contents on the Web, using a simple high-level schema notation (e.g. OPM or Genera's own notation). Genera programs process this description to generate database query commands and formatting instructions that together extract objects from the database and format them into HTML documents on demand. Genera also supports form-based querying and whole-database formatting into text and HTML formats.

SUMMARY OF THE GDB 6.0 SCHEMA

Figure 2 shows a simplified view of the object class hierarchy for the Genome Data Base version 6.0. *DBObject*, the root class, contains basic attributes pertinent to all GDB objects, such as owner, release date and accession number. The remainder of the important objects in the database are divided among five core classes:

GenomeObjects

Things making up or associated with genomes, such as *GenomeRegions*, *GeneFamilies* and *GeneProducts*. *GenomeRegion* is an enhancement of the concept of *Locus* from previous GDB releases. It includes chromosomes, genes, phenotypic markers, cytogenetic landmarks, STS's and contigs, among others.

MapObjects

Data that describe order and distance relations among regions of the genome, as inferred from various mapping experiments. Other classes in this category represent higher-order relationships (i.e. alignments) between maps.

ExperimentObjects

Information about mapping experiments, experimental reagents and experimental results from observed interactions between reagents.

VariationObjects

Data describing mutations, polymorphisms, population frequencies and so on.

AnnotationObjects

Objects that allow users of the database to comment on other objects in GDB. Literature citations, annotations and cross-references to other databases may be associated with anything else in the database. Citations are shown with a dashed box in Figure 2 to indicate that they are actually stored in a separate database (see below).

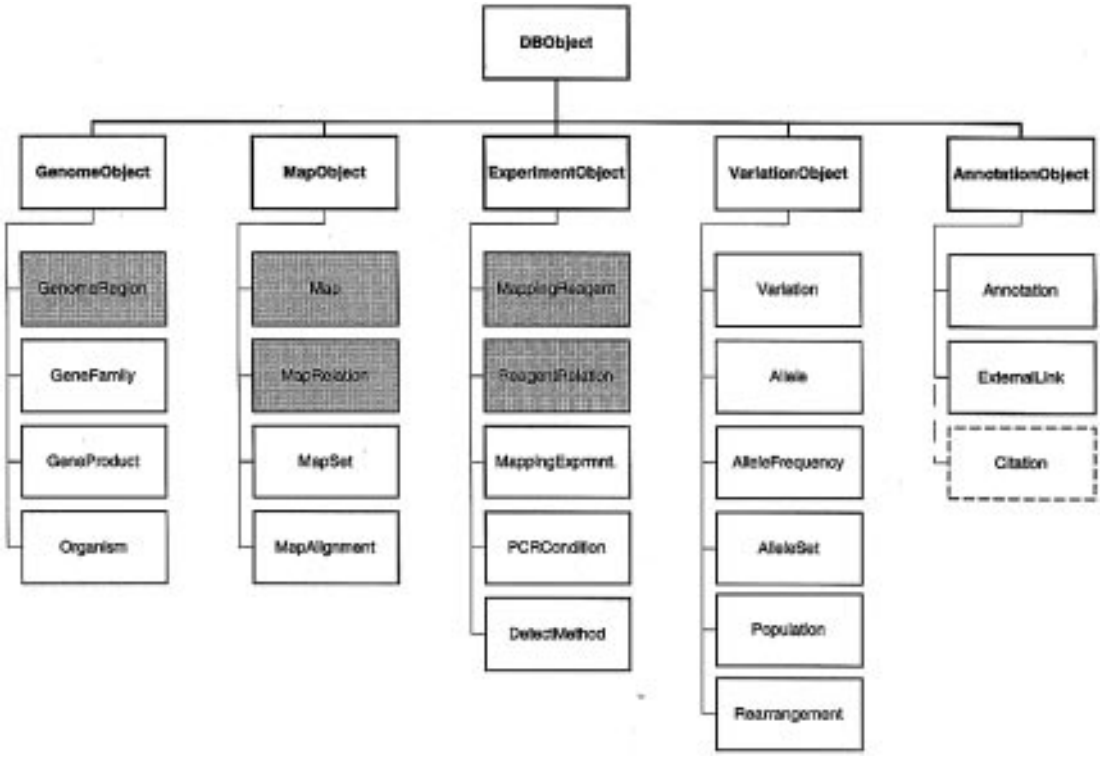

Figure 2. Simplified view of the object class hierarchy for GDB 6.0. Objects in the database are divided among five classes: *GenomeObject, MapObject, ExperimentObject, VariationObject* and *AnnotationObject*, all of which inherit basic bookkeeping attributes (name, owner, etc.) from the *DBObject* class. The shaded classes represent the correspondence between experimental entities and inferred genomic relationships. *Citations*, though actually stored in a separate database, are treated like other annotation objects. A number of minor administrative classes have been left out of this diagram for clarity.

In addition to those shown in Figure 2, there are several other types of object classes in the new GDB design. These are 'lightweight' objects, controlled vocabularies and administrative objects. Lightweight objects take their name from the fact that they have many fewer bookkeeping attributes than do *DBObjects*. Objects in the database that do not require accession numbers, annotation, support for confidential submission and so forth are grouped in this category. Controlled vocabularies are classes containing standardized attribute descriptions, such as amino acid names, developmental stages and map distance units. They are essential in any good database design for insuring a uniform representation of the data for query and analysis purposes. Administrative objects are classes in the GDB 6.0 schema that support the internal workings of the database, such as GDB staff annotations and accession number allocation. Controlled vocabularies are editable by HUGO committees and GDB staff, but not the general public. Administrative objects are only editable by GDB staff.

GDB 6.0 is part of a family of interrelated data sets operated by the Genome Data Base project. The mapping database described here is known as Human Genome Database (HGD) within the context of this 'mini-federation'. The other components are CitDB, which holds literature citations and the Genome Registry, which holds information on people and organizations in the genome community. We view the separation of this information into multiple databases as a pilot project for a subsequent effort to federate genomic databases across the Internet.

Detailed documentation on the latest database schema can be obtained from GDB's WWW and anonymous ftp servers (see below). This is part of our continuing effort to have an open, interactive dialogue with the community concerning GDB's content and implementation. We welcome feedback from genome researchers on this and all aspects of the Genome Data Base's design and operations.

A FAREWELL TO D-NUMBERS?

The concept of anonymous DNA segments ('D-segments') and their associated 'D-number' identifiers, was introduced to describe standard genomic landmarks that could be mapped, but whose role in the genome was otherwise uncharacterized (9,10). Originally, D-segments were defined by regions where anonymous clones had been mapped to a single chromosome or better. Since clone names were not unique or stable, official D-number assignments provided a robust means of referring to the same anonymous markers. Later, after the concept of the sequence tagged site was introduced (11), mapped STS's were assigned D-numbers as well. Thus began the problem that has been debated between genetic and physical mappers ever since.

There is no utility for genetic mappers in assigning multiple D-numbers which cannot be distinguished at typical linkage mapping resolution. As genetic mapping predominated prior to and in the early days of the Human Genome Project, the early history of D-segment allocation involved the not-infrequent

merging of D-segments when they were found to map to the same location. This posed a problem for physical mappers, since with their resolving power the corresponding clones or STS's could be easily separated. D-segments could not be reliably placed on physical maps if their true physical extent was allowed to change over time.

In July of 1992, the Genome Data Base convened an *ad hoc* advisory group to address this problem (12). The group consisted of members of the HUGO DNA Committee and other physical and genetic mappers. Lengthy debate resulted in a complex plan to refine D-segment nomenclature to distinguish between loosely defined anonymous regions and well localized landmarks such as STS's. A version of this plan were presented to the full body of HUGO committee members at the 1992 Chromosome Coordinating Meeting (CCM92, ref. 13) and was tabled, largely due to its complexity. In their CCM93 report, the DNA Committee proposed that 'each physically definable segment of DNA (i.e. cloned insert or an STS) which is used in a physical mapping study [is] to receive its own DNA segment assignment. This method of assignment will continue until a revised definition of DNA segments [can] be proposed, which would more adequately address the needs of both the genetic and physical mapping communities.' (14).

It is our belief that GDB 6.0 provides an opportunity to address this issue once and for all. The first step was actually provided in version 5.1 with the introduction of accession numbers ('GDBid's') for all objects in the database. GDBid's can take the place of D-numbers completely. More importantly, because the concept of *GenomeRegion* in GDB 6.0 is a significant improvement over *Locus* in previous GDB versions, genomic regions such as STS's and cloned segments can now be rigorously defined and properly localized in genetic and physical maps. Since genome regions can be referred to as points or intervals depending on the overall mapping resolution, they may participate as appropriate in both types of maps. Based on this, the 'Genome Data Base would like the human genome community to strongly consider the use of GDBid's in place of D-numbers. We would particularly urge journal editors, HUGO committees, single chromosome workshop and other conference organizers to insist on the presence of GDB accession numbers in all abstracts and publications, accompanying traditional D-numbers if not actually in lieu of them'.

While Genome Data Base staff will continue to assign D-numbers upon request according to DNA Committee guidelines, we hope that the scientific community will come to see the advantages of using GDBid's in their place.

PLANNED ENHANCEMENTS

A number of additional features are planned for near-term releases of the Human Genome Data Base. These will be described briefly below.

Graphical map editor

The MapViewer will be extended to support direct entry and editing of GDB maps. The proposed map editor, already under development, will simplify interactive preparation of manuscript figures and GDB data submissions. It will allow existing GDB maps to be modified with immediate graphical feedback. Subsequent versions will incorporate techniques from SIGMA (15) to facilitate viewing and editing of very large maps.

Integrated editing, querying and browsing

An integrated suite of programs for editing, querying and browsing GDB 6.0 is already in development. When completed in the spring of 1996, it will provide a more sophisticated environment for creating new submissions to the database or modifying existing ones. Unlike the Genera-based interface, the GDB application suite will also allow data submissions to be prepared off-line, for subsequent transmission to Baltimore via e-mail, anonymous FTP or posted diskette.

Integration with GSDB Annotator

We are currently working with Genome Sequence Data Base to integrate their Sequence Annotator program with the GDB program suite so that researchers will be able to view and edit maps and sequences together.

Improved integration with the Mouse Genome Database

The goals of the Mouse Genome Database (16) and the Human Genome Data Base are extremely similar; the two databases manage much of the same types of information. More importantly, the utility of comparative analysis of the mouse and human genomes is well recognized. While links already exist between human genes in GDB and homologous loci in MGD, further integration of the two data sets is desirable. For example, we are working to develop map viewers for comparison of conserved syntenies in the two genomes.

Improved polymorphism and mutation representation

The Human Genome Project seems to be making a transition from 'one-pass' analysis of the genome to a more detailed categorization of variation. Witness the rapid growth of specialty databases for cataloguing the known mutations at important loci (e.g. 17,18). While GDB already has a significant body of data on human polymorphisms, it has only a modest amount of mutation information. We plan to enhance our database representations for both rare and common variation and to work in conjunction with the curators of mutation databases and sequence databases to increase GDB's utility as a central repository for this information.

SUMMARY OF GDB SERVICES

The Genome Data Base provides the following services:

WWW. http://gdbwww.gdb.org/

Anonymous ftp. ftp://ftp.gdb.org/,

Data files, documentation, standard reports, software.

Electronic mail server. mailserv@gdb.org,

Retrieve data based on simple keyword search through an e-mail query system. For information, include the word 'help' in the body of the message to this e-mail address:

WAIS. wais://wais.gdb.org/

WAIS sources include data organized as flat files: cell line, citation, contact, library, locus, map, mutation, polymorphism and probe.

CONTACTING GDB

Baltimore

Questions about database content or obtaining a user account should be directed to: GDB User Services, Johns Hopkins University School of Medicine, 2024 E. Monument St, Suite 1–200, Baltimore, MD 21205-2100, USA, Tel: +1 410 955 9705, FAX: +1 410 614–0434, E-mail: help@gdb.org.

Data contribution

For information regarding the submission of data to GDB, address inquiries to Data Acquisition and Curation at the above mailing address, telephone and fax numbers or preferably via e-mail to: data@gdb.org.

GDB international sites

The Genome Data Base provides access to the database at the following international nodes:
Australia. ANGIS, University of Sydney, bucholtz@angis.su.oz.au; WEHI, Walter and Eliza Hall Institute, Melbourne, tony@wehi.edu.au.
France. INSERM, Villejuif, gdb@infobiogen.fr.
Germany. DKFZ, Heidelberggdb@dkfz-heidelberg.de.
Israel. Weizmann Institute of Science, Rehovot, lsprilus@weizmann.weizmann.ac.il.
Japan. JICST, Tokyomika@gdb.gdbnet.ac.jp.
The Netherlands. CAOS/CAMM, University of Nijmegen, Nijmegen, post@caos.caos.kun.nl.
Sweden. Uppsala Biomedical Center, Uppsala, help@gdb.embnet.se.
UK. HGMP Resource Center, Hinxton, admin@hgmp.mrc.ac.uk.

CITING THE GENOME DATA BASE

When citing the Genome Data Base in the literature, please reference this article as: Fasman, K.H., Letovsky, S.I., Cottingham, R.W. and Kingsbury, D.T. (1996) Improvements to the GDB™ Human Genome Data Base. *Nucleic Acids Res.*, **24**, 57–63.

ACKNOWLEDGEMENTS

The GDB Human Genome Data Base is an international project funded by a grant from the US Department of Energy (DE-FC02-9ER6130) with additional support from the US National Institutes of Health, the Science and Technology Agency of Japan, the Medical Research Council of the UK, the INSERM of France and the European Union.

REFERENCES

1 Pearson, P.L. (1991) *Nucleic Acids Res.*, **19**, 2237–2239.
2 Pearson, P.L., Matheson, N.W., Flescher D.C. and Robbins R.J. (1992) *Nucleic Acids Res.*, **20**, 2201–2206.
3 Cuticchia, A.J., Fasman, K.H., Kingsbury, D.T., Robbins, R.J. and Pearson, P.L. (1993) *Nucleic Acids Res.*, **21**, 3003–3006.
4 Fasman, K.H., Cuticchia, A.J. and Kingsbury, D.T. (1994) *Nucleic Acids Res.*, **22**, 3462–3469.
5 Stephens, J.C., Cohen, I.H. and Kidd, K.K. (1990) The Human Gene Mapping Library: present status and future directions. In Bell, G. and Marr, T. (eds), *Computers and DNA, Santa Fe Institute Studies in the Sciences of Complexity.* Vol. VII. Addison-Wesley, Redwood City, CA.
6 Chen, I.A. and Markowitz, V.M. (1995) *Information Systems*, **20**, 393–418. URL - http://gizmo.lbl.gov/DM_TOOLS/OPM/opm.html.
7 Letovsky, S. and Berlyn, M. (1992) *Genomics*, **12**, 435–446.
8 Letovsky, S.I. Genera. URL - http://gdbdoc.gdb.org/letovsky/genera/genera.html.
9 Gusella, J.F., Keys, C., Varsanyi-Breiner, A., Kao, F.-T., Jones, C., Puck, T.T. and Housman, D. (1980) *Proc. Natl. Acad. Sci. USA*, **77**, 2829–2833.
10 Skolnick, M.H. and Francke, U. (1982) *Cytogenet. Cell Genet.*, **32**, 194–204.
11 Olson, M., Hood, L., Cantor, C. and Botstein, D. (1989) *Science*, **245**, 1434–1435.
12 Bakker, B., Bowcock, A., Ceverha, P., Chipperfield, M., Fasman, K., Green, P., Kidd, K., Klinger, K. and Pearson, P. (1992) Report of the GDB Advisory Group for DNA Nomenclature Issues. GDB internal report.
13 Bowcock, A.M., Klinger, K., Bakker, E., Pearson, P.L., Chipperfield, M.A., Ceverha, P. and Minter, A. (1993) Report of the DNA committee - Including highly informative markers. *Genome Priority Rep.*, **1**, 810–884.
14 Bowcock, A.M., Bakker, E., Ceverha, P., Chipperfield, M.A., Minter-Morrison, A., Porter, C.J. and Pearson, P.L. (1994) Report of the DNA committee. In Cuticchia, A.J. and Pearson, P.L. (eds), *Human Gene Mapping 1993, A Compendium.* Johns Hopkins University Press, Baltimore, MD, pp. 893–972.
15 National Center for Genome Resources. SIGMA: System for integrated genome map assembly. On-line manual. URL - http://www.ncgr.org/sigma.
16 Mouse Genome Database (MGD), Mouse Genome Informatics Project, The Jackson Laboratory, Bar Harbor, Maine. URL - http://www.informatics.jax.org.
17 Tuddenham, E.G.D., Schwaab, R., Seehafer, J., Millar, D.S., Gitschier, J., Higuchi, M., Bidichandani, S., Connor, J.M., Hoyer, L.W., Yoshioka, A., Peake, I.R., Olek, K., Kazazian, H.H., Lavergne, J.-M., Giannelli, F., Antonarakis, S.E. and Cooper, D.N. (1994) *Nucleic Acids Res.*, **22**, 3511–3533.
18 Hollstein, M., Rice, K., Greenblatt, M.S., Soussi, T., Fuchs, R., Sørlie, T., Hovig, E., Smith-Sørensen, B., Montesano, R. and Harris, C.C. (1994) *Nucleic Acids Res.*, **22**, 3551–3555.