Flexible Genome Retrieval for Supporting In-Silico Studies of Endobacteria-AMFs

S. Montani[1], G. Leonardi[1], S. Ghignone[2], and L. Lanfranco[2]

[1] Dipartimento di Informatica, University of Piemonte Orientale, Alessandria, Italy
[2] Dipartimento di Biologia Vegetale, University of Turin, Italy

Abstract. Studying the interactions between arbuscular mycorrhizal fungi (AMFs) and their symbiotic endobacteria has potentially strong impacts on the development of new biotechnology applications. The analysis of genomic data and syntenies is a key technique for acquiring information about phylogenetic relationships and metabolic functions of such organisms.

In this paper, we describe a modular architecture meant to support in-silico genome sequence analysis, which is being developed within the project BIOBITS. In particular, we focus on a flexible genome retrieval tool, which supports optimized and customized comparative genomics searches.

1 Introduction

Arbuscular mycorrhizal fungi (AMFs) are obligate symbionts which, to complete their life cycle, must enter in association with the root of land plants. Here, they become a crucial component of soil microbial communities, and exert positive impacts on plants health and productivity: in particular, they furnish a better mineral nutrition, and act as crucial means for increasing tolerance to stress conditions [13,19].

AMFs are thus a significant resource for sustainable agriculture; in addition, they could also be exploited as a still unknown resource to promote green (agriculture) and white (industrial) biotechnologies. For instance, plants release molecules (strigolactones) which are perceived by AMFs, and cause the extensive branching essential for a successful colonization. Similar molecules have a large relevance in chemistry, since they induce seed germination from parasite plants, like Striga, which infest the two-thirds of crop lands in Africa.

AMFs are often in further symbiosys with uncultivable bacteria, living inside the AMF itself [4]. The resulting tripartite system (i.e. (i) endobacterium; (ii) AMF; (iii) plant roots) is a complex biological object, whose extensive study requires a comparative genomics approach, in order to answer fundamental questions concerning the biology, ecology and evolutionary history of the system and of its composing elements. As a matter of fact, comparative genomics represent a key instrument to discover or validate phylogenetic relationships, to give insights on genome evolution, and to infer metabolic functions of a given organism, which is particularly useful when biochemical and physiological data are not available and/or hard to obtain.

H. Takeda (Ed.): E-Health 2010, IFIP AICT 335, pp. 138–147, 2010.

Studying the tripartite system has potentially strong practical impacts, given the assumption that the symbiotic consortia may lead to new metabolic pathways, and to the appearance of molecules which might be of interest for biotechnological applications.

While bacterial endosymbionts in the animal kingdom are excellent models for investigating important biological events, such as organelle evolution, genome reduction, and transfer of genetic information among host lineages [17], examples of endobacteria living in fungi are limited [12]. A key part of the study about the tripartite system mentioned above is therefore represented by the analysis of the genomic data of the endobacteria themselves. In particular, large-scale analysis and comparison of genomes belonging to phylogenetically related free-living bacteria can provide information about the events that led to genome down-sizing, and insights about the reason of the strict endosymbiotic life style of the bacteria themselves.

In this paper, we present the design of a computational environment for the genomic study of the AMF Gigaspora margarita (isolate BEG34) and of its endobacterium *Candidatus Glomeribacter gigasporarum*, which are currently used as a model system to investigate endobacteria-AMFs interactions. In particular, we are developing a modular architecture, composed by a database, in which massive genomic data are imported and stored, and of genomic comparison (synteny) and visualization tools.

To achieve such an aim we plan to exploit as much as possible the available tools built around GMOD, the Generic Model Organism Database project [18], which brought to the development of a whole, and still under expansion, collection of open source software tools for creating and managing genome-scale biological databases. In particular, we are choosing to resort to the GMOD database Chado [9], and to some freely available and freely adaptable GMOD facilities to search for syntenies like CMap, GBrowse_syn, SyBil.

However, we are extending the functionalities offered by GMOD, in order to properly meet the needs of our specific comparative genomics research. In particular, we are working at a *flexible* similar genomes retrieval tool, implementing the *retrieval* step of Case-Based Reasoning (CBR) [1], an Artificial Intelligence methodology which supports human reasoning by recalling past experiences similar to the current one. Our tool allows to search in the genomes database by expressing queries at *different levels of detail*, also in an interactive fashion. Moreover, it takes advantage of *multi-dimensional orthogonal index structures*, which make retrieval faster, allowing for early pruning and focusing.

The work, which is still in its early implementation phase, is supported by the BIOBITS project, a grant of Regione Piemonte, under the Converging Technologies Call, which involves the University of Turin, the University of Piemonte Orientale, the CNR and the companies ISAGRO Ricerca s.r.l., GEOL Sas, Etica s.r.l.

The paper is organized as follows: section 2 provides a deeper description of the biological problem under examination; section 3 sketches the general architecture we are implementing for genome analysis in the project; section 4 focuses on our flexible retrieval tool, and section 5 is devoted to conclusions.

2 Description of the Biological Domain

AMF species, belonging to the family Gigasporaceae, represent a specialized niche for rod shaped bacteria, which due to their current unculturable status have been grouped into a new taxon named *Candidatus Glomeribacter gigasporarum* [8]. The AM fungus Gigaspora margarita (isolate BEG34) and its endobacterium *Ca. Glomeribacter gigasporarum* are currently used as a model system to investigate endobacteria-AM fungi interactions. Microscopical observations have shown that the endobacteria are Gram-negative, rod-shaped, approx. 0.8-1.2 µm by 1.5-2.0 µm; they occur singly or in groups and are often inside fungal vacuoles. The analysis of the 16S ribosomal RNA sequence demonstrated that these bacteria are phylogenetically related to the beta-proteobacterial family, clustering with the Burkholderia, Pandorea and Ralstonia genera [6, 12]. Morphological and molecular studies have shown that *Ca. G. gigasporarum* is a homogeneous population, which is vertically transmitted through the fungal generations [7]. The main limitation to studying *Ca. Glomeribacter gigasporarum* is that it has not been isolated in pure culture even after testing several growth conditions [10]; therefore, this bacterium is considered an obligate endocellular component of its fungal host. A sufficient amount of endobacterial DNA was obtained from fungal spores and used to estimate a genome size of about 1.3 Mb, depending on the method used, consisting of a chromosome and a plasmid [10]. This small genome size is consistent with a strict endosymbiotic nature. Indeed small chromosomes have only been encountered in other well known obligate endocellular species [21, 4, 23, 16]. Because *Ca. G. gigasporarum* does not grow in pure culture, traditional genetic or physiological studies can not be applied. Therefore, the analysis of the genome sequence may offer a valuable tool to infer its metabolic functions; at the moment the bacterial genome is being sequenced by two complementary strategies, classical Sanger sequencing of fosmid clones combined to high-throughput pyrosequencing using the 454 platform (Roche).

3 System Architecture

In our project, genome sequence analysis is supported by a modular architecture, which permits:

(1) to store and access locally all the information regarding the organisms to be studied, and
(2) to provide algorithms and user interfaces to support the researchers' activities (e.g.: search and retrieval of genomes, comparison and alignment with a reference genome, investigation of syntenies and local storage of potential new annotations).

The system architecture has been engineered exploiting the standard modules and interfaces offered by the GMOD project [18], and completed with custom modules to provide new functionalities (see Figure 1).

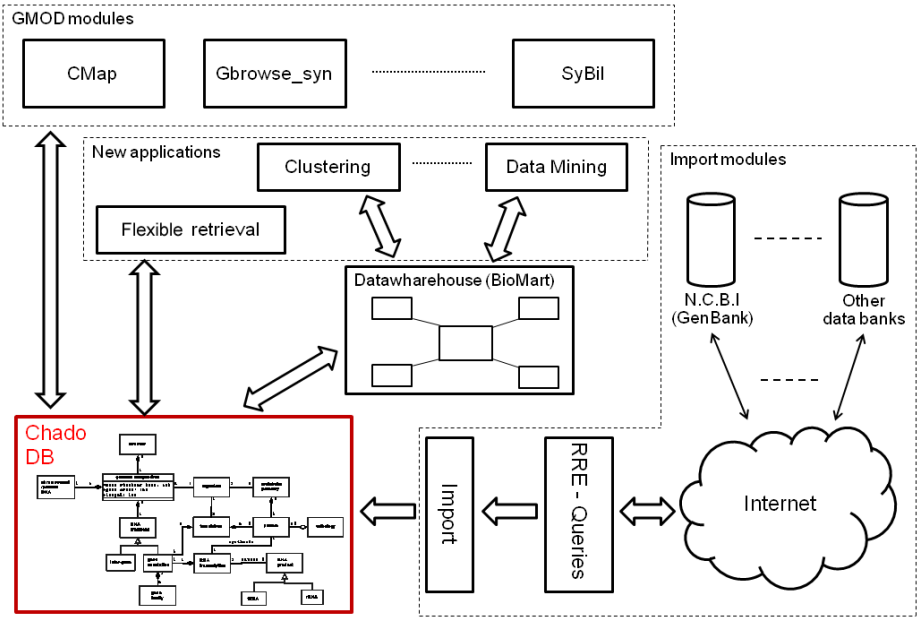

Fig. 1. The system architecture

The main module of the system contains the database which provides all the data needed to perform the in-silico activities. We adopted the Chado database schema [9], to take advantage of its completeness and of its support for controlled vocabularies and ontologies. Furthermore, Chado is the standard database for most of the GMOD modules, therefore we can reuse these modules to support the main activities of the project, and extend the system incrementally as the researchers' needs evolve. The database in this module stores and provides all the information about the organisms to be studied (i.e. bacteria), their genomes, their known annotations, their proteins and metabolic pathways, and the newly discovered annotations, which can be stored and managed locally until they are confirmed and published.

As explained, our database contains information to be used and stored locally, but we have added the possibility to populate and update the database with information retrieved from the biological databases accessible through the Internet. This feature is provided by the set of modules in the *Import modules* section (see Figure 1). The main module (*RRE - Queries*), which is built on the basis of a previously published tool [11], performs queries to different biological databases through the Internet (e.g. the GenBank [26]) and converts the results into a standard format. Afterwards, the *Import* module inserts or updates the retrieved information into the Chado database. This process can be started on-demand, or performed automatically on a regular basis, in order to maintain the local database up-to-date.

Chado also acts as the data interface for the software layers implementing the functionalities and tools used by the researchers. From the architectural point of view, we offer two types of services: the services implemented through existing modules of

GMOD (*GMOD modules* section in Figure 1), and new services implemented through new modules, developed ad-hoc (*New applications* section). The latter is composed, at the time of writing, of:

- a module called *BioMart* [25], which reorganizes the information stored in the Chado database into a data warehouse, in order to analyze the data by means of clustering and other data mining techniques (whose description is outside the scope of this paper) and
- a *flexible retrieval* module, described in section 4, which supports efficient retrieval strategies in the context of the search for genomic similarity and syntenies.

The *GMOD modules* section exploits the available GMOD modules using the Chado database to provide the researchers with the tools for comparative genomics needed in the BIOBITS project. In particular:

- CMap allows users to view comparisons of genetic and physical maps. The package also includes tools for maintaining map data;
- GBrowse is a genome viewer, and also permits the manipulation and the display of annotations on genomes;
- GBrowse_syn is a GBrowse-based synteny browser designed to display multiple genomes, with a central reference species compared to two or more additional species;
- SyBil is a system for comparative genomics visualizations.

All the tools in this system use a web-based interface to be more user-friendly and easy to use. Many GMOD modules can be reused as they are, but they can be customized to meet the researchers' recommendations before being integrated in our software architecture. Furthermore, every new module added in the *New application*s section of our architecture, or every customized module in the *GMOD modules* section, connects to the other modules of our architecture using GMOD standard interfaces. Therefore, every new or customized module can be published to the GMOD community, in order to extend and enrich this platform.

4 Flexible Retrieval of Similar Genomes

The *flexible retrieval* module we have designed in the project architecture implements the *retrieval* step of the Case-Based Reasoning (CBR) [1] cycle. CBR is a reasoning paradigm that exploits the knowledge collected on previously experienced situations, known as *cases*. The CBR cycle operates by:

(1) *retrieving* past cases that are similar to the current one and by
(2) *reusing* past successful solutions after, if necessary, properly
(3) *adapting* them; the current case can then be
(4) *retained* and put into the system knowledge base, called the *case base*.

Purely retrieval systems, leaving to the user the completion of the reasoning cycle (steps 2 to 4), are however very valuable decision support tools [24], especially when

automated adaptation strategies can hardly be identified, as in biology and medicine [14]; in the project, we are following this research line.

In our module, cases are genomes as sequences of nucleotides, each one taken from a different organism, and properly aligned with the same reference organism. For each nucleotide, a percentage of similarity with the aligned nucleotide in the reference organism is also provided.

However, depending on the type of analysis which is required, a "view" of the genomes at the nucleotide level may not always be the most appropriate: sometimes, a "higher level" view, abstracting the available data at the level of genes, regions, or even complete chromosomes, would be more helpful. Our tool supports this need, by allowing the retrieval of the available cases at any level of detail, according to a taxonomy of granularities, which is depicted in figure 2.

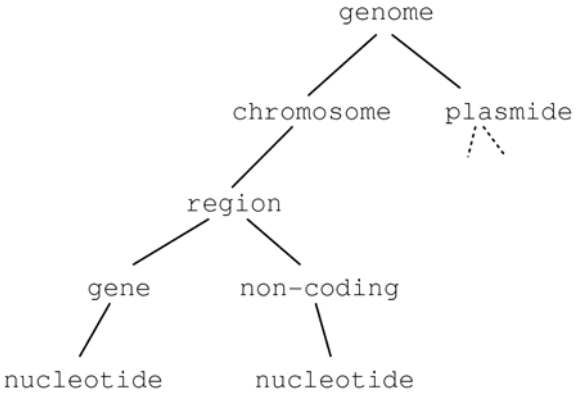

Fig. 2. A taxonomy of sequence granularities

Moreover, a sequence of consecutive granules, sharing the same *qualitative level* (e.g. low, medium, high) of similarity with respect to the reference organism, can be abstracted into a single interval, labeled with the qualitative level of similarity itself: such an abstraction process is very similar to the Temporal Abstractions (TA) methodology, described in [20,5], even if in our domain the independent variable is the granules sequence instead of time. As in TA, in fact, we move from a *point-based* to an *interval-based* representation of the data, where the input points are the granules, and the output intervals (*episodes*) aggregate adjacent points sharing a common behavior, persistent over the sequence. In particular, we rely on *state* abstractions [5], to extract episodes associated with qualitative levels of similarity with the reference organism, where the mapping between qualitative abstractions and quantitative values (percentages) of similarity can be parameterized on the basis of domain knowledge.

Space occupancy in the database can be optimized by storing the abstracted data instead of the original ones. Moreover, on abstracted data, case retrieval can benefit from the use of pattern matching techniques (see e.g. [22]).

Also in the state abstractions dimensions, we allow the user to express her queries at different levels of detail, depending on her current analysis interests, according to a

state abstractions taxonomy like the one described in figure 3 - which can be properly modified depending on specific domain needs.

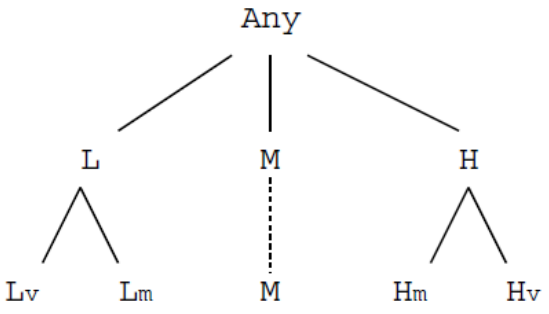

Fig. 3. An example taxonomy of state abstraction symbols; for instance, the high (H) symbol specializes into very high (Hv) and moderately high (Hm)

In synthesis, our retrieval framework allows for *multi-level abstractions*, according to two *dimensions*, namely a taxonomy of state abstraction symbols, and a variety of sequence granularities. In particular, we allow for *flexible querying*, where queries can be expressed at any level of detail in both dimensions.

Moreover, our framework takes advantage of *multi-dimensional orthogonal index structures*, which make retrieval faster, allowing for early pruning and focusing. The root node of each index structure is represented by a (string of) symbol(s), defined at the highest level in the state abstraction taxonomy (i.e. the children of *Any*, see figure 3) and in the sequence granularity taxonomy. A (possibly incomplete) index stems from each root, describing possible refinements along the symbol and/or the sequence granularity dimension. An example multi-dimensional index, rooted in the H symbol, is represented in figure 4.

Each node in each index structure is itself an index, and can be defined as a *generalized case*, in the sense that it summarizes (i.e. it indexes) a set of cases. This means that the same case is typically indexed by different nodes in one index (and in the other available indexes). This supports flexible querying, since, depending on the level at which the query is issued, according to the two taxonomies, one of the nodes can be more suited for providing a quick answer.

To answer a query, in order to enter the more proper index structure, we first progressively generalize the query itself in the state abstractions taxonomy direction, while keeping sequence granularity fixed. Then, we generalize the query in the granularity dimension as well. Following the generalization steps backwards, we can enter the index from its root, and descend along it, until we reach the node which fits the original query sequence granularity. If an orthogonal index stems from this node, we can descend along it, always following the query generalization steps backwards. We stop when we reach the same detail level in the state abstraction taxonomy as in the original query. If the query detail level is not represented in the index, because the index is not complete, we stop at the most detailed possible level. We then return all the cases indexed by the selected node.

Interactive and *progressive* query relaxation or refinement are supported as well in our framework, in a conversational fashion [2]. Query relaxation (as well as refinement) can be repeated several times, until the user is satisfied with the width of the retrieval set.

The interested reader may find additional technical details in [15]. Very encouraging experimental results have already been obtained by resorting to the same flexible retrieval framework, in the field of haemodialysis [15].

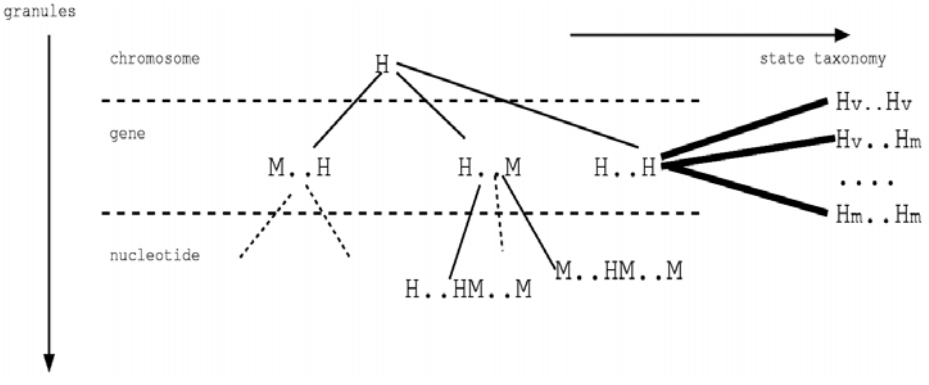

Fig. 4. An example multi-dimensional orthogonal index. Note that indexes may be incomplete with respect to the taxonomies: here, for instance, the region level is missing in the granularity dimension.

5 Conclusions

In this paper, we have described a modular architecture for supporting in-silico comparative genomics analysis, being developed within the BIOBITS project. In particular, we have focused on the main features of a genome retrieval tool, which implements the first step of the CBR cycle. Such a tool provides researchers with flexible retrieval capabilities, also in an interactive fashion. Moreover, retrieval performances are optimized by resorting to multi-dimensional orthogonal index structures, allowing for a quick query answering. In the next months, we will complete the tool implementation, and work at the collection of the first evaluation results, initializing the tool case base with a selection of genomes (in particular, from symbiotic microorganisms) available in the RefSeq NCBI database.

References

1. Aamodt, A., Plaza, E.: Case-based reasoning: foundational issues, methodological variations and systems approaches. AI Communications 7, 39–59 (1994)
2. Aha, D., Munoz-Avila, H.: Introduction: interactive case-based reasoning. Applied Intelligence 14, 78 (2001)

3. Akman, L., Rio, R.V.M., Beard, C.B., Aksoy, S.: Genome size determination and coding capacity of sodalis glossinidius, an enteric symbiont of tsetse flies, as revealed by hybridization to escherichia coli gene arrays. J. Bacteriol. 183, 4517–4525 (2001)
4. Anca, I.A., Lumini, E., Ghignone, S., Salvioli, A., Bianciotto, V., Bonfante, P.: The ftsZ gene of the endocellular bacterium 'Candidatus Glomeribacter gigasporarum' is preferentially expressed during the symbiotic phases of its host mycorrhizal fungus. Molecular plant-microbe interactions: MPMI 22(3), 302–310 (2009)
5. Bellazzi, R., Larizza, C., Riva, A.: Temporal abstractions for interpreting diabetic patients monitoring data. Intelligent Data Analysis 2, 97–122 (1998)
6. Bianciotto, V., Bandi, C., Minerdi, D., Sironi, M., Tichy, H.V.: An obligately endosymbiotic fungus itself harbors obligately intracellular bacteria. Appl. Environ. Micro biol. 62, 3005–3010 (1996)
7. Bianciotto, V., Genre, A., Jargeat, P., Lumini, E., Baecard, G., Bonfante, P.: Vertical transmission of endobacteria in the arbuscular mycorrhizal fungus gigaspora margarita through generation of vegetative spores. Appl. Environ. Microbiol. 70, 3600–3608 (2004)
8. Bianciotto, V., Lumini, E., Bonfante, P., Vandamme, P.: Candidatus glomeribacter gigasporarum, an endosymbiont of arbuscular mycorrhizal fungi. Int. J. Syst. Evol. Microbiol. 53, 121–124 (2003)
9. Mungall, C.J., Emmert, D.B.: The FlyBase Consortium. A chado case study: an ontology-based modular schema for representing genome-associated biological information. Bioinformatics 23(13), i337–i346 (2007)
10. Jargeat, P., Cosseau, C., Olah, B., Jauneau, A., Bonfante, P.: Isolation, free-living capacities, and genome structure of candidatus glomeribacter gigasporarum the endocellular bacterium of the mycorrhizal fungus gigaspora margarita. J. Bacteriol. 186, 6876–6884 (2004)
11. Lazzarato, F., Franceschinis, G., Botta, M., Cordero, F., Calogero, R.: Rre: a tool for the extraction of non-coding regions surrounding annotated genes from genomic datasets. Bioinformatics, 1 20(16), 2848–2850 (2004)
12. Lumini, E., Ghignone, S., Bianciotto, V., Bonfante, P.: Endobacteria or bacterial endosymbionts? to be or not to be. New Phytol. 170, 205–208 (2006)
13. Marx, J.: The roots of plant-microbe collaborations. Science 304, 234–236 (2004)
14. Montani, S.: Exploring new roles for case-based reasoning in heterogeneous AI systems for medical decision support. Applied Intelligence 28, 275–285 (2008)
15. Montani, S., Bottrighi, A., Leonardi, G., Portinale, L., Terenziani, P.: Multi-level abstractions and multi-dimensional retrieval of cases with time series features. In: McGinty, L., Wilson, D.C. (eds.) ICCBR 2009. LNCS, vol. 5650, pp. 225–239. Springer, Heidelberg (2009)
16. Moran, N.A., Dale, C., Dunbar, H., Smith, W.A., Ochman, H.: Intracellular symbionts of sharpshooters (insecta: Hemiptera: Cicadellinae) form a distinct clade with a small genome. Env. Microbiol. 5, 116–126 (2003)
17. Moran, N.A., McCutcheon, A.J., Nakabachi, P.: Genomics and evolution of heritable bacterial symbionts. Annu. Rev. Genet. 42, 165–190 (2008)
18. Brian Osborne and GMOD Community. GMOD (2000)
19. Parniske, M.: Arbuscular mycorrhiza: the mother of plant root endosymbioses. Nat. Rev. Microbiol. 6, 763–775 (2008)
20. Shahar, Y.: A framework for knowledge-based temporal abstractions. Artificial Intelligence 90, 79–133 (1997)

21. Shigenobu, S., Watanabe, H., Hattori, M., Sakaki, Y., Ishikawa, H.: Genome sequence of the endocellular bacterial symbiont of aphids buchnera sp aps. Nature 407, 81–86 (2000)
22. Stephen, G.A.: String searching algorithms. Lecture Notes Series in Computing, vol. 3. World Scientific, Singapore (1994)
23. Sun, L.V., Foster, J.M., Tzertzinis, G., Ono, M., Bandi, C., Slatko, B.E., ONeill, S.L.: Determination of wolbachia genome size by pulsed-field gel electrophoresis. J. Bacteriol. 183, 2219–2225 (2001)
24. Watson, I.: Applying Case-Based Reasoning: techniques for enterprise systems. Morgan Kaufmann, San Francisco (1997)
25. http://www.biomart.org/
26. http://www.ncbi.nlm.nih.gov/Genbank/