Facing the Challenges of Genome Information Systems: A Variation Analysis Prototype

Ana M. Martínez, Ainoha Martín, Maria José Villanueva,
Francisco Valverde, Ana M. Levin, and Oscar Pastor

Centro de Investigación en Métodos de Producción de Software
Universidad Politécnica de Valencia
Camino de Vera S/N 46022, Valencia, Spain
{amartinez,amartin,mvillanueva,fvalverde,alevin,opastor}@pros.upv.es
http://www.pros.upv.es

Abstract. In Bioinformatics there is a lack of software tools that fit with the requirements demanded by biologists. For instance, when a DNA sample is sequenced, a lot of work have to be performed manually and several tools are used. The application of Information Systems (IS) principles into the development of bioinformatics tools opens a new interesting research path. One of the most promising approaches is the use of conceptual models in order to precisely define how genomic data is represented into an IS. This work introduces how to build a Genome Information System (GIS) using these principles. As a first step to achieve this goal, a conceptual model to formally describe genomic mutations is presented. In addition, as a proof of concept of this approach, a variation analysis prototype has been implemented using this conceptual model as a development core.

1 Introduction

In 1953, James D. Watson and Francis Crick described the DNA structure as a "double helix"[1]. In 1990, the Human Genome Project [2] officially begins and twenty-three years later, in 2003, all the effort is rewarded and the whole sequence appears [3,4]. The availability of this new information opens the door to the creation of new ways of diagnosis, new types of medicines, new therapy strategies, new studies, etc.

Thanks to this breakthrough and the advances in DNA sequencing, a big amount of genetic data is being produced by researchers every day. Most of these experiments are focused on the understanding of the relationship between genotype (gene configuration and combination of a particular individual) and its phenotype (expression of the genes in a specific human feature). As a consequence, the creation of biological databases and tools to exploit the data produced has grown drastically. However, these tools and databases have usually been defined to support a specific research area or experiment. Therefore, when biologists want to use them for a particular essay, it is very unlikely that they support their specific requirements.This leads to a situation where the researcher

P. Soffer and E. Proper (Eds.): CAiSE Forum 2010, LNBIP 72, pp. 222–237, 2010.

has to spend a lot of time and effort to perform a simple analysis. Since these bioinformatics tools are not developed using IS principles, they are not aligned with the user requirements. The main consequences of this issue are:

- Some biological databases are only human readable, thus cannot be processed properly in an automatic way.
- The extraction of relevant data is difficult because it is spread around different databases.
- Since several tools are required to analyze the data, the tooling workflow specification and integration is far from trivial.
- Inclusion of new studies and bibliography into the available tools turns into a hard task.

To solve these questions, some researchers have proposed [5] the development of Genomic Information Systems (GIS), specifically designed IS capable to handle a big amount of genomic data. In this work, a new approach to develop GIS is proposed: the use of conceptual models to define and organize the genomic data in a formal way.

Thanks to the close collaboration with biologists in the context of this work, the gap between the disciplines of software engineering and genetics can be solved. The result of this interdisciplinary collaboration is the design of a conceptual model that guides the alignment of concepts among both fields. Therefore the design and implementation of the software artifacts that made up a GIS becomes an easier process.

Following that idea, this paper presents a GIS prototype that analyzes DNA sequences and compares them to the reference in order to find variations for a specific gene. Once all variations are located in the sequence, the prototype splits them into two groups: one group contains harmless variations and the other one contains variations that produce a change in gene or protein function. For those in the last group, their specific phenotype is reported as it has been described in the literature.

This information is bibliographically referenced and gathered in a report that helps the researcher to understand the genetic meaning of the variation and why it produces a certain phenotype. This is very useful because it can speed up the diagnosis of a specific disease. Furthermore, it is widely accepted that an early detection of a disease might be determinant.

The rest of the paper is organized as follows. In section 2 a review of DNA variation analysis tools is presented. Section 3 details a conceptual model to describe genomic mutations. Section 4 describes how the variation analysis prototype has been developed. Finally, in section 5 conclusions and future work are stated.

2 Related Work

There are two approaches to perform a DNA sequence analysis: genotyping, which analyzes small DNA fragments, and sequencing, which analyzes the whole

genome. In this section, some tools that use these approaches are analyzed. The majority of the studied tools use genotyping, which means that in the majority of the cases relevant information located at unexplored regions is ignored. To solve this problem, the presented prototype uses the sequencing approach to detect all the variations produced in one gene.

In recent years, several commercial tools have been developed to provide genomic analysis. These tools perform tests to estimate the probability of the customer to suffer certain diseases. Navigenics [6],23andMe [7], deCODEme [8], DNADirect [9] and Knome [10] are the most relevant tools in this field. The differences between them are briefly summarized in Table 1. 23andMe, Navigenics, DNADirect and deCODEme are tools that help their clients to make decisions about their health providing information such as: their probability of suffering certain disease, the mutations that could affect their family future and its own or the possibility of finding what pharmacotherapy is best suited for their organism. These mentioned services use genotyping to analyze the sequence. This approach does not analyze the whole sequence, only small fragments, as for example SNPs (Single Nucleotide Polymorphism -variation that occurred at least in 0,5 or 1% on the population-), which are interpreted to perform the diagnostic.

As a result of using genotyping, the analysis provided may miss some relevant data set in these unexplored areas. And in general, the diagnosis obtained from these tools should be always supervised by a doctor.

Other drawback of these tools is that the only variations reported are SNPs. The prototype improves the quality of this analysis detecting other complex variations, such as insertions or deletions. Furthermore, the diseases detected by these commercial tools are limited to the number of supported genes. The prototype overcomes this constraint since the conceptual model supports the addition of new genes and their discovered variations.

Table 1. Comparison of DNA analysis tools

	Navigenics	23andMe	deCODEme	Knome	DNADirect
Analysis Type	Genotyping	Genotyping	Genotyping	Sequencing	Sequencing
Platform	Affymetrix [11]	Illumina [12]	Illumina [12]	SOLID3	Illumina[12]
Variations (10^6)	1 (only SNP)	0.5 (only SNP)	1.2 (only SNP)	unlimited	-
Detected diseases	28	51	49	+1000	-

On the other hand, Knome and DNADirect are equivalent tools that offer a revolutionary approach using sequencing instead of genotyping. Sequencing reduces the limitations of genotyping approaches, and enables the detection of other variants that cannot be identified with the above commented tools. However, these tools are far from been accesible to the people due to its price.

The complete understanding of the genomic field can only be achieved through the integration of biological information and the different analysis tools and applications available [13]. The proliferation of these tools has thus increased the importance of workflow systems.

In the commercial field it is possible to find workflow systems such as Pipeline Pilot [14], the first workflow system in life sciences. Pipeline Pilot is widely applied in drug discovery and high-throughput screening (HTS), but it also encompasses Decision Trees, Squene Analysis, BioMining, Text Analytics and Integration Collection, which are flexible mechanisms to link external applications and databases. Another commercial workflow system is the one of InforSense KDE [15] which has got specialized extensions such as BioSense, ChemSense and TextSense. The first mentioned extension covers high performance bioinformatics solutions ranging from sequence analysis to microarrays informatics and remote database annotation.

Both Pipeline Pilot and InforSense KDE help researchers in the life sciences domain in a fast and efficient way. Nevertheless this help is diminished because they cost a lot of money. Therefore these tools are still not accessible to academics and small labs that cannot invest so much money for that kind of solutions.

On the other hand, there are a lot of open-source workflow systems, some of them began as small-scale projects. The major part of the workflow systems in the public domain differ in their architecture and in other features such as workflow language, primary data types, mechanism to add additional resources and available domain resources.

The open-source workflow systems represent an important aid for the scientific domain not only because they are free, but also because they are founded on community development models, in which people from different backgrounds have contributed to the application in an active way. It is worth mention that there are many commercial products that make use of open-source and public available programs. An example of an open-source workflow system is Pegasys [16], developed by the university of British Columbia. This specialized workflow management gives high-throughput sequence data analysis and annotation. Pegasys also includes numerous tools for pair-wise and multiple sequence alignment, gene prediction, RNA gene detection and masking repetitive sequences in genomic DNA, and it also permits the incorporation of new tools into existing frameworks due to its flexible architecture. ^{my}Grid project is considered the most powerful workflow system in the public domain. Fields like Genome and Proteome Annotation (e-Protein), Integrative Systems Biology (myIB) or Integration of Biological Data (PlaNet, EMBRACE) are covered by different tools based on ^{my}Grid. But the most important workflow system rooted on ^{my}Grid is Taverna. The use of booth tools has been extensively employed to execute different *in silico* experiments like simple sequence manipulation or genotype-phenotype correlations. The Genome Analysis and Database Update (GADU) system uses Pegasus to perform high-throughput analysis and annotation of genomic information. GADU workflows are being run across the Open Science Grid and TeraGrid, and to enrich the warehouse they are applying tools such as BLAST or BLOCKS. Nevertheless, the disadvantage of Pegasus is that it does not support a validation checking for workflow.

On the other hand, due to the information explosion in biology, the number of ontologies grows directly proportional to the biological data. The classical

example of an ontology created to provide a list of controlled terms to describe biological entities is the Gene Ontology [17]. The GO Consortium was established to create standard terms to describe what the gene products do, where they act and how they perform these activities. Another example is the one of BioPAX [18] whose purpose is to provide a standard language that enables integration, exchange, visualization and analysis of biological pathways data. TAMBIS project [19] includes an ontology, the TAMBIS ontology, that aims to provide transparent information retrieval and filtering from biological information services by building a homogenizing layer on top of the different sources.

Nevertheless, the purpose of an ontology is to give a description of the terminology used in a specific domain rather than to provide a conceptual representation of the structures used to store data [20]. Contrary to conceptual schemas, ontologies do not allow showing the data structure and relations. Thus the comprehension of the domain is easier in conceptual schemas. The way in which data can be described is made explicit by separating the information models from the system description, which is an advantage for developers of future GIS. For instance, the information technology developers can quickly and easily adopt a model to implement a new feature in a domain without having to repeatedly tackle the same complex modeling challenges. The developed system will be interoperable with other projects that use the same conceptual model. Therefore, tasks such as data integration from different repositories will be simplified. There are some projects that are developing this idea; one is the Phenotye and Genotype Experiment Object Model [21] which has been approved as a standard by OMG. The PaGE-OM was formulated by an international consortium of 20 groups involved in genotype-phenotype projects. The goals of PaGE-OM specification are to reach a balance between being too generic and too specific and to enable the structured capture of at least the minimum amount of the information required to properly report most generic experiments related to genotype and phenotype information. [20] provides conceptual models that describe eukaryotic genome sequence data and genome organization, transcription data and results from gene deletions. Other object-oriented models of biological data have been implemented, such as the one presented in [22], which includes models for representing genomic sequence data.

3 A Conceptual Model to Describe Variations

The main objective of the conceptual model presented in this paper is to establish a connection between the genomic field and the IS development domain. One of the main characteristics of the genomic field is heterogeneity. The unification of relevant concepts is a difficult task, since genomic concepts are not precisely defined. Moreover the field of knowledge is still developing and concepts are constantly evolving, wich complicates the organization of all the genetic data available.

Genetic databases are affected by this heterogeneity problem. In this fielda, each database captures the concepts according to the interpretation and terminology of a biologist. However, there are different definitions for the same

concept; for example, a variation in the DNA sequence is referred under the terms: variation, mutation, polymorphism or SNP [23]. Even though all of them represent more or less the same concept, there are slight differences among them.The problem of heterogeneous data representation can be solved with the use of conceptual models, as some works propose [24]. The development of a conceptual model to represent the human genome is a useful approach to understand this complex domain since precise concepts are defined and related among them. If new concepts, relations or changes are discovered, they can be easily incorporated on the model.

The conceptual model presented here claims to be precise with genetic concepts and IS principles because it has been developed by software engineers and biologists specialized in the genomic field. The model presented in this section is focus on the description of genomic variations. However, it is an excerpt of a widest one [25], whose main goal is the specification of the required human genome concepts for developing GIS.

Figure 1 shows the proposed conceptual model. At the top of the picture (1) the *Gene* and the *Allele* classes are defined. *Gene* entity models the generic concept of gene whereas *Allele* entity represents the individual instances of a gene. The *Allele* entity has two specializations: *Allelic Reference Type* and *Allelic Variant*. *Allelic Reference Type* models the reference sequence that defines an "universal" gen to be used for comparison purposes. These reference sequences are extracted from trusted data sources as RefSeqGene database [26]. *Allelic Variant* represents a DNA sequence of an individual which has several variations from the allelic reference.

Each variation discovered by means of the comparison process performed over a sequence, is modeled by the *Variation* entity (2). The *Variation* entity stores all the variations documented in the genetic literature that are associated to some disease or to normal changes because of the intrinsic nature of an individual. This entity has two different specializations groups. The first one corresponds to the variation description and it is made up of two specializations: *Precise* variations, which define variations that are completely located and *Imprecise* variations, whose location details are not specified. *Precise* variations are also categorized in four entities according to the change performed in the sequence: *Insertion*, *Deletion*, *Indel* (insertion/deletion) and *Inversion*. An indel can be categorized as *SNP* as well when it occurs at least in 1% of the population. The second group consists of three specializations: *Mutant* variations, which represents those variations that are related to some disease, *Unknown consequence*, categorizes the specializations whose consequence has not been discovered yet, and *Neutral polymorphism* variations, the ones that do not have an associated disease.

A variation that is specified in the model is always related to its phenotype, which is modeled by the *Phenotype* entity (4). The *Certainty* entity specifies the probability that a phenotype could show up because of a concrete variation on the genotype. In case a genotype-phenotype association is identified, it is essential to know information about the bibliographic reference and the original database where the discovery was stated. This data is defined by the *Bibliography* Reference

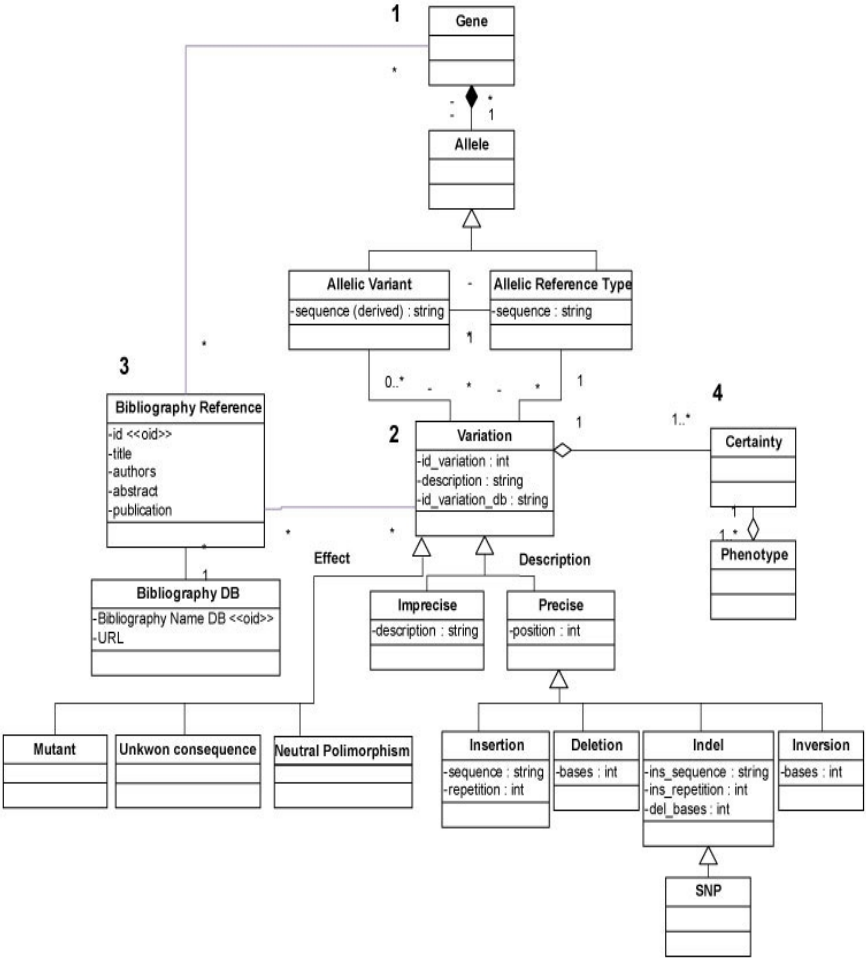

Fig. 1. Conceptual Model for describing variations

and *BibliographyDB* entities (3) respectively. As a first result of this conceptual model, a genetic database (GDB) has been created to store the variation information that is used by the presented GIS prototype.

4 A GIS Proof of Concept : A Variation Analysis Tool

The main goal of the prototype is to show how conceptual models can be useful to define a GIS. One of the most common tasks in the genomic area is the analysis of genomic sequences [27]. Researchers perform the analysis comparising a certain DNA sample from a specific gene and its reference sequence. The comparison is done using an alignment tool that shows a list of differences among them. After that, an experienced researcher has to decide which variations are relevant and which not. Afterwards, they have to dive into the vast and

non-structured amount of information that is scattered across the Web and search for the bibliography that justifies each relevant variation. Performing this work manually is a tedious and time consuming task.

The proposed prototype reduces this time by automating the major part of the manual work. This automation can be done thanks to the conceptualization of the domain by the presented conceptual model. Data such as genes, variations, phenotypes and bibliographic references are now represented as perfectly defined conceptual entities. Thanks to this conceptualization, heterogeneity and data dispersion problems are solved, avoiding the manual preprocess of some non-computer legible data and ensuring the quality of the data stored.

The most widely accepted and used free implementation of BLAST [28] is NCBI BLAST, but more sequence alignment algorithms exist. One of them is BLAT [28], which is an extremely fast alternative but slightly less accurate than BLAST to compare huge nucleotide sequences. This is the chosen algorithm for the implementation of the prototype proposed in this paper.This choice is based on its speed to scan for relatively short matches and also its extension into high-scoring pairs. Differing from BLAST [29], BLAT stitches each area of homology between two sequences into a larger alignment. The prototype explained in this section has been implemented as a web tool developed using ASP.NET and C#.

The service offered by the presented GIS prototype is to receive a DNA sample from a patient and provide a report that helps the doctor to diagnose a certain disease. The experts only have to introduce the sample in the suitable format and review the provided results, forgetting everything about manual treatment and endless searches across the bibliography.

The analysis process performed by the prototype is summarized in Figure 2. Some conceptual model entities that are used in the different steps are depicted in white rectangular boxes. The process is divided into five main steps:

1. Input data: The biologist selects a gene from the set supported by the prototype, for instance the BRCA1 gene, and introduces the DNA sample to be analyzed. The sample input can be performed manually or by uploading a file in FASTA format.
2. Alignment report: According to the selected gene, the prototype locates the suitable reference using the allelic reference entity. After that, an alignment process between the sample and the reference is carried out for finding variations. This alignment is performed using the BLAST algorithm, however importing results from DNA sequencing tools as Sequencher [13] will be supported in next versions. Using the defined conceptual model, each discovered difference is formalized as an instance of the variation entity. This formalization, which is not present at the moment in other tools or databases, is independent of the output from any alignment tool and provides a suitable way for exchanging variations. A report that summarizes all the changes is generated using these variation entities.
3. Variation knowledge: Thanks to the report generated in the previous phase the classification problem is simplified. Variations are located according to a well-know reference sequence and their positions matched to the genomic

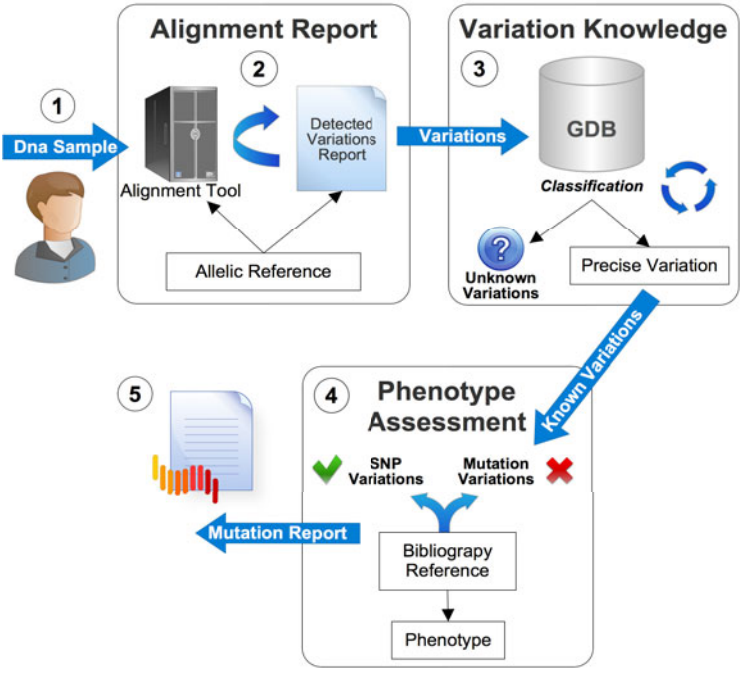

Fig. 2. Mutational analysis tool based on the conceptual model

data stored in GBD. Then, each variation is queried into the GDB to determine if it has been defined as a precise variation. If a variation cannot be found in our GDB is classified as unknown. At this point, known variations are classified into a specific type of sequence change. Unknown variations are classified as non-silent if the variation produces a change, in other words, an effect in the expected gene product (protein).

4. Phenotype Assessment: Variations classified as known may have some phenotype associated. In order to asses if the phenotype is related to a specific disease, a research publication is required to provide trustful evidence. For those cases, the conceptual model describes the bibliographical reference that supports the phenotype for a specific variation. In the context of this work, variations with pathogenic phenotype are classified as mutations whereas they are classified as SNPs if no negative phenotype is described.
5. Report creation: All the obtained information is gathered in a report. This report contains information about the variations found: mutations, variations whose phenotype is not a disease and unknown variations. Each variation is provided with the following information: the location where it was found in the sequence, its type (Insertion, Deletion, Indel or Inversion) and the number of nucleotides inserted or deleted. For the mutations found in the GDB their associated phenotype and its bibliography is added as well. Finally, the report file can be saved as a text document.

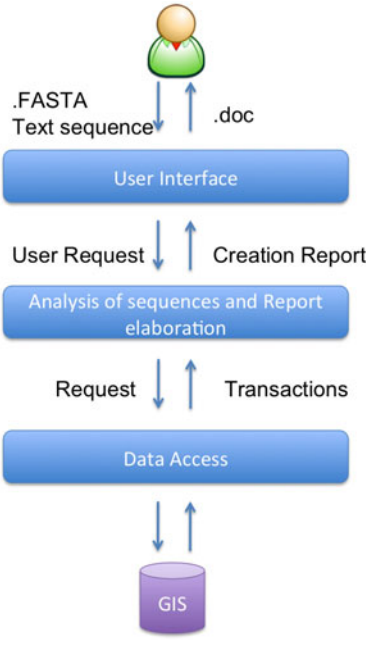

Fig. 3. Architecture of the variation analysis tool

Several key points have been discussed during the implementation of this prototype. The first one was the choice of the architecture to be implemented. A multi-layer architecture is the one that suits best to carry out the implementation of the prototype proposed here. This type of architecture allows a logical separation of the processes related to the presentation, the application processing, and the data management. N-tier application architecture provides a model for developers to create a flexible and reusable application.

By breaking up an application into layers, developers only have to modify or add a specific tier, rather than rewrite the entire application over. Having into account that in the biological domain, and specially in the genomic field, a lot of investigations are being carried out and thus new information is being discovered, it is important to facilitate the way in which the developed system can be modified and maintained. The three-layer architecture implemented to this application is showed in Figure 3.

Another issue is to define the classes that are going to be part of the application. Figure 4 shows a brief summary of the most important implemented classes of the variation analysis tool and also some of their attributes and services.

It's important to keep in mind that the goal of this prototype is to obtain all the mutations that belong to a certain DNA sequence. Some of the differences between this class diagram and the model represented in Figure 1 are a direct consequence of this constraint.

Variation is the principal class. It represents the variations that are found after the DNA sequence analysis. Due to the comparison between DNA sequences,

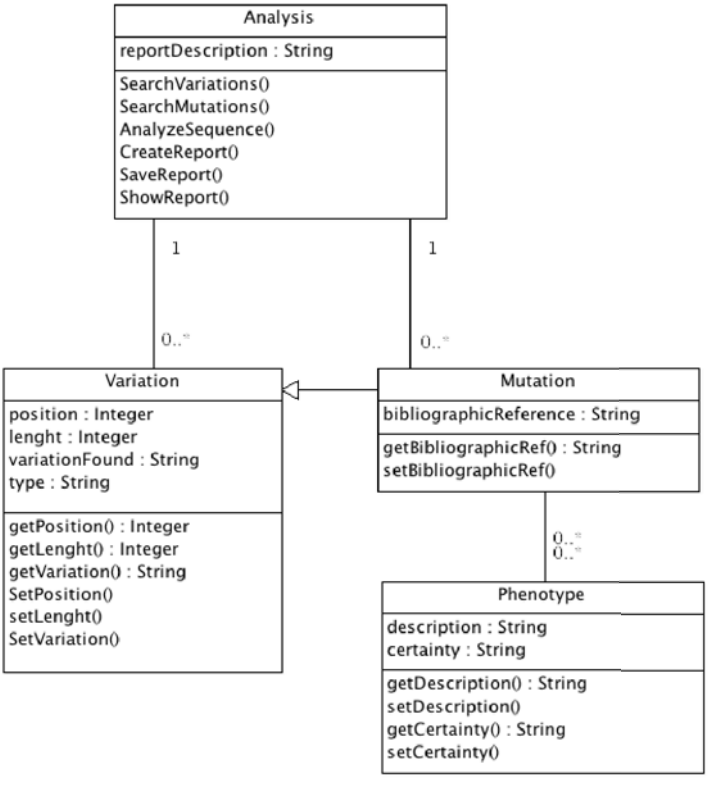

Fig. 4. Diagram Class

only specific positions are obtained. Thus, only precise variations are analyzed and imprecise variations remain outside of the application scope at this moment. Hence, *Variation* class and *Precise* class are joined in one unique class. The specializations of *Precise* class are also joined to this class becoming the type attribute. These classes define the difference between the types of variations, and their attributes can be unified in the same way as they did. Therefore, these new attributes are added to Variation class to facilitate the implementation of the tool by unifying the way in which the properties of the variation type are represented. The services of this class are more related to access and manage their own properties.

Mutation class represents the variations that are related to a disease. *Mutations* class has got the same main properties as *Variation* so, it is implemented as a specialization. Nevertheless, there are some attributes of the class *Mutation* but not of *Variation*, for instance the *BibliographicReference* attribute. As mentioned before, the implemented tool is focused on the mutation search and that is the reason why only this kind of variation needs a bibliographic reference. Instead of using a new class to represent this property, the bibliographic reference is implemented as a *Mutation* class attribute. This choice is made based on the

Fig. 5. GIS prototype interface

assumption that this kind of information is going to be read only as a property of a variation and not managed. In fact, the method *setBibliographicRef* is private and it is only used to provide the value of this attribute through the data base extraction method.

The *Phenotype* class represents the way a variation in a DNA sequence is expressed. However, taking into account the constraint commented at the beginning, this class is only linked to *Mutation* class. *Phenotype* has, apart from its description, an attribute certainty that express the reliability of the phenotype.

Finally the *Analysis* class is the main class of the application and it is responsible for the report creation. The visibility of the *SearchVariations*, *SearchMutations*, *AnalyzeSequence* and *ShowReport* methods is private. The user can only make use of the *CreateReport* and *SaveReport* methods and these services will be responsible for calling the private ones internally.

Referring to the visual aspect Figure 5 shows the interface of the presented GIS prototype. This interface is divided into three parts:

– Input data: At the top there is a list of the currently supported genes. The user can choose which gene is the object of study. There is a text area below to choose the reference sequence. Additionally, the researcher is provided

with two ways to insert the patient's DNA sequence. The first one is a text area where the researcher can introduce the DNA sequence directly. The second way is by uploading a FASTA file.
- Functionality: The actions provided are three: 1) showing the report, this action takes place after the data entry and works in background without the researcher being aware of it 2) cleaning the work area, clean button allows to clear the fields where the researcher can enter data 3) and saving the information obtained.
- Medical report: this is the area in which all the information obtained by the tool is displayed to the user. The different features that were commented in previous sections: location, position, type of variation, etc., are listed here.

5 Conclusions and Future Work

In the genetics domain, the occurrence of a new database is a constant fact. There are a lot of genomic databases containing different kinds of data (variations of DNA sequences, gene specific disease information, ARN or protein data, pathways, microarrays, etc.). Despite the distinct nature of their content, all of them were built under a common goal: structuring data in order to achieve a better comprehension of their discoveries. However, most of them are really useful for the daily work of researchers; but when dififcult questions arise, the problem of interrelate different data located in several databases appear.

This work proposes an IS engineering solution in order to solve the heterogeneity problems on the genomic domain. A conceptual model is presented which describes and defines formally the concepts related to genomic variations. As a proof of concept, a GIS prototype with this conceptual model as background has been implemented. This prototype analyzes human DNA samples searching for variations and identifies the phenotypes that each variation could provoke on the individual.

One of the advantages of using the presented GIS prototype is that the variation analysis can be performed using only one tool, avoiding the data workflow. In addition, using a conceptual model to guide the development simplifies the acquisition of the genetic data and is precisely referenced to the bibliography.

However, the study of the prototype efficiency working with real DNA samples must be analyzed. In order to perform this task, further studies of the sequencing algorithms will be carried out.

Moreover, heterogeneity is not a completely novel research area because some tools to organize the genomic data have also been proposed before [17,20,30]. The main contribution of the work presented here is a conceptual model specifically designed to guide the development of software artifacts using a model-driven approach.

As further work it is planned to extend the GIS prototype in order to achieve a higher accuracy and to facilitate the sequence input. As a final goal, the GIS prototype will be tested in a real environment by means of a collaboration with IMEGEN, a genomic medicine institute, and a couple of local hospitals.

References

1. Watson, J., Crick, F.: A structure for deoxyribose nucleic acid. Nature 171, 737–738 (1953)
2. Jordan, E.: The American Journal of Human Genetics 51, 1–6 (1992)
3. Craig, J., Venter, J.C., Adams, M.D., Myers, E., Li, P.W., Mural, R.J., Sutton, G.G., Smith, H.O., Yandell, M., Evans, C.A., Holt, R.A., Gocayne, J.D., Amanatides, P., Ballew, R.M., Huson, D.H., Wortman, J.R., Zhang, Q., Kodira, C.D., Zheng, X.H., Chen, L., Skupski, M., Subramanian, G., Thomas, P.D., Zhang, J., Gabor Miklos, G.L., Nelson, C., Broder, S., Clark, A.G., Nadeau, J., McKusick, V.A., Zinder, N., Levine, A.J., Roberts, R.J., Simon, M., Slayman, C., Hunkapiller, M., Bolanos, R., Delcher, A., Dew, I., Fasulo, D., Flanigan, M., Florea, L., Halpern, A., Hannenhalli, S., Kravitz, S., Levy, S., Mobarry, C., Reinert, K., Remington, K., Abu-Threideh, J., Beasley, E., Biddick, K., Bonazzi, V., Brandon, R., Cargill, M., Chandramouliswaran, I., Charlab, R., Chaturvedi, K., Deng, Z., Di Francesco, V., Dunn, P., Eilbeck, K., Evangelista, C., Gabrielian, A.E., Gan, W., Ge, W., Gong, F., Gu, Z., Guan, P., Heiman, T.J., Higgins, M.E., Ji, R.R., Ke, Z., Ketchum, K.A., Lai, Z., Lei, Y., Li, Z., Li, J., Liang, Y., Lin, X., Lu, F., Merkulov, G.V., Milshina, N., Moore, H.M., Naik, A.K., Narayan, V.A., Neelam, B., Nusskern, D., Rusch, D.B., Salzberg, S., Shao, W., Shue, B., Sun, J., Wang, Z., Wang, A., Wang, X., Wang, J., Wei, M., Wides, R., Xiao, C., Yao, A., Ye, J., Zhan, M., Zhang, W., Zhang, H., Zhao, Q., Zheng, L., Zhong, F., Zhong, W., Zhu, S., Zhao, S., Gilbert, D., Baumhueter, S., Spier, G., Carter, C., Cravchik, A., Woodage, T., Ali, F., An, H., Awe, A., Baldwin, D., Baden, H., Barnstead, M., Barrow, I., Beeson, K., Busam, D., Carver, A., Center, A., Cheng, M.L., Curry, L., Danaher, S., Davenport, L., Desilets, R., Dietz, S., Dodson, K., Doup, L., Ferriera, S., Garg, N., Gluecksmann, A., Hart, B., Haynes, J., Haynes, C., Heiner, C., Hladun, S., Hostin, D., Houck, J., Howland, T., Ibegwam, C., Johnson, J., Kalush, F., Kline, L., Koduru, S., Love, A., Mann, F., May, D., McCawley, S., McIntosh, T., McMullen, I., Moy, M., Moy, L., Murphy, B., Nelson, K., Pfannkoch, C., Pratts, E., Puri, V., Qureshi, H., Reardon, M., Rodriguez, R., Rogers, Y.H., Romblad, D., Ruhfel, B., Scott, R., Sitter, C., Smallwood, M., Stewart, E., Strong, R., Suh, E., Thomas, R., Tint, N.N., Tse, S., Vech, C., Wang, G., Wetter, J., Williams, S., Williams, M., Windsor, S., Winn-Deen, E., Wolfe, K., Zaveri, J., Zaveri, K., Abril, J.F., Guigó, R., Campbell, M.J., Sjolander, K.V., Karlak, B., Kejariwal, A., Mi, H., Lazareva, B., Hatton, T., Narechania, A., Diemer, K., Muruganujan, A., Guo, N., Sato, S., Bafna, V., Istrail, S., Lippert, R., Schwartz, R., Walenz, B., Yooseph, S., Allen, D., Basu, A., Baxendale, J., Blick, L., Caminha, M., Carnes-Stine, J., Caulk, P., Chiang, Y.H., Coyne, M., Dahlke, C., Mays, A., Dombroski, M., Donnelly, M., Ely, D., Esparham, S., Fosler, C., Gire, H., Glanowski, S., Glasser, K., Glodek, A., Gorokhov, M., Graham, K., Gropman, B., Harris, M., Heil, J., Henderson, S., Hoover, J., Jennings, D., Jordan, C., Jordan, J., Kasha, J., Kagan, L., Kraft, C., Levitsky, A., Lewis, M., Liu, X., Lopez, J., Ma, D., Majoros, W., McDaniel, J., Murphy, S., Newman, M., Nguyen, T., Nguyen, N., Nodell, M., Pan, S., Peck, J., Peterson, M., Rowe, W., Sanders, R., Scott, J., Simpson, M., Smith, T., Sprague, A., Stockwell, T., Turner, R., Venter, E., Wang, M., Wen, M., Wu, D., Wu, M., Xia, A., Zandieh, A., Zhu, X.: The Sequence of the Human Genome Science 291, 1304–1351 (2001)
4. Collins, F.S., Green, E.D., Guttmacher, A.E., Guyer, M.S.: A vision for the future of genomics research Nature 422, 835–847 (2003)

5. Gilbert, D.G.: Eugenes: a eukaryote genome information system. Nucleic Acids Research 30, 145–148 (2002)
6. Navigenics (2010), http://www.navigenics.com
7. 23andme (2010), https://www.23andme.com
8. Decodeme (2010), http://www.decodeme.com
9. Medco acquires leading genetics healthcare company, DNA Direct (2005), http://www.dnadirect.com/web/
10. Knome (2010), http://www.knome.com
11. Irizarry, R.A., Bolstad, B.M., Collin, F., Cope, L.M., Hobbs, B., Speed, T.P.: Summaries of affymetrix genechip probe level data. Nucleic Acids Research 31, e15 (2003)
12. Klein, R.: Power analysis for genome-wide association studies. BMC Genetics 8, 58 (2007)
13. Tiwari, A., Sekhar, A.K.: Workflow based framework for life science informatics. Computational Biology and Chemistry 31, 305–319 (2007)
14. Hassan, M., Brown, R.D., Varma-O'brien, S., Rogers, D.: Cheminformatics analysis and learning in a data pipelining environment. Molecular Diversity 10, 283–299 (2006)
15. Watson, C., Guo, Y., Sheldonb, J.: Yike Guo and Jonathan Sheldon of InforSense discuss the impact of workflow technology on drug discovery. Drug Discovery Today 10, 1211–1212 (2005)
16. Shah, S., He, D., Sawkins, J., Druce, J., Quon, G., Lett, D., Zheng, G., Xu, T., Ouellette, B.: Pegasys: software for executing and integrating analyses of biological sequences. BMC Bioinformatics 5 (2004)
17. Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., Harris, M.A., Hill, D.P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J.C., Richardson, J.E., Ringwald, M., Rubin, G.M., Sherlock, G.: Gene ontology: tool for the unification of biology. Nature Genetics 25, 25–29 (2000)
18. BioPax-Consortium: Biological pathways exchange (2005), http://www.biopax.org/
19. Stevens, R., Baker, P., Bechhofer, S., Ng, G., Jacoby, A., Paton, N.W., Goble, C.A., Brass, A.: Tambis: Transparent access to multiple bioinformatics information sources. Bioinformatics 16, 184–186 (2000)
20. Paton, N.W., Khan, S.A., Hayes, A., Moussouni, F., Brass, A., Eilbeck, K., Goble, C.A., Hubbard, S.J., Oliver, S.G.: Conceptual modelling of genomic information. Bioinformatics 16, 548–557 (2000)
21. Brookes, A., Lehvaslaiho, H., Muilu, J., Shigemoto, Y., Oroguchi, T., Tomiki, T., Mukaiyama, A., Konagaya, A., Kojima, T., Inoue, I., Kuroda, M., Mizushima, H., Thorisson, G., Dash, D., Rajeevan, H., Darlison, M.W., Woon, M., Fredman, D., Smith, A.V., Senger, M., Naito, K., Sugawara, H.: The phenotype and genotype experiment object model (PaGE-OM): a robust data structure for information related to DNA variation. Human Mutation 30, 968–977 (2009)
22. Medigue, C., Rechenmann, F., Danchin, A., Viari, A.: Imagene: an integrated computer environment for sequence annotation and analysis. Bioinformatics 15, 2–15 (1999)
23. den Dunnen, J.T., Antonarakis, E.: Nomenclature for the description of human sequence variations. Human Genetics 109, 121–124 (2001)
24. Richesson, R., Turley, J.P.: Conceptual models: Definitions, construction, and applications in public health surveillance. Journal of Urban Health 80, i128 (2006)

25. Pastor, O., Levin, A., Casamayor, J., Celma, M., Virueta, A., Eraso, L., Pérez-Alonso, M.: Enforcing conceptual modeling to improve the understanding of human genome. In: Procs. of the IVth Int. Conference on Research Challenges in Information Science (2010)
26. NCBI: The refseqgene project (2010), http://www.ncbi.nlm.nih.gov/RefSeq/RSG
27. Stevens, R., Goble, C., Baker, P., Brass, A.: A classification of tasks in bioinformatics. Bioinformatics 17, 180–188 (2001)
28. Kent, W.J.: Blat, the blast-like alignment tool. Genome Research 12, 656–664 (2002)
29. Altschul, S., Gish, W., Miller, W., Myers, E.W., Lipman, D.: Basic local alignment search tool. Journal of Molecular Biology 215, 403–410 (1990)
30. Ram, S.: Toward Semantic Interoperability of Heterogeneous Biological Data Sources. In: Pastor, Ó., Falcão e Cunha, J. (eds.) CAiSE 2005. LNCS, vol. 3520, pp. 32–32. Springer, Heidelberg (2005)