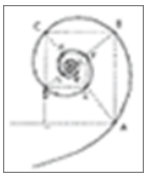
Università degli Studi di Milano
Scuola di Dottorato in Medicina Molecolare



Curriculum di Genomica, Proteomica e Technologie Correlate

Ciclo XXV

Anno Accademico 2012-2013

Clinical genomic research management

Dottorando: Amitava BHATTACHARYYA
Matricola: R08878

Tutore: Prof. Daniele CUSI

Direttore della Scuola: Ch.mo Prof. Mario CLERICI

Dedicated to the noble soul … who perished!

Abstract

Technological advancement in Genomics has propelled research in a new era, where methods of conducting experiments have completely been renovated. Riding the wave of Information Technology, equipped with statistical tools, Genomics provide a revolutionized perspective unthought-of in the past. With the completion of the Human Genome project, we have a common reference for analysis at the level of the complete genome. High throughput technologies for gene expression, genotyping and sequencing are propelling present research. Attempts are now being made for the incorporation of these methods in the health care in a structured format. Clinicians cherish the use of genomics for the assessment disease predisposition and realizing personalized medical care for a better health care. As genome sequencing is becoming swifter and its cost reducing, the public genomic data has increased many folds. Data from other high throughput technologies and annotations further increase the storage requirements. Laboratory management software, LIMS, is now becoming the limiting factor as automation and integration increases. Thus genomics now faces the challenge of management of this enormous data catering to varied needs, not limited only for the research laboratories, but extends also to health care institutions and individual clinicians. Further, there is a growing need for the analysis and visualization of the generated data to be integrated into the same platform for a continuous research experience and systematic supervision. Data security is of prime concern, especially in health care concerning human subjects. The interest of the clinicians adds another management requirement, a delivery system for the concerned subject.

Hypertension is a complex disorder with world-wide prevalence. HYPERGENES project was centered on the objective of integrating biological data and processes with Hypertension as the disease model. The HYPERGENES project focuses on the definition of a comprehensive genetic epidemiological model of complex traits like Essential Hypertension (EH) and intermediate phenotypes of hypertension such as Target Organ Damage (TOD). During the HYPERGENES project, the above mentioned challenges were comprehended and evaluated, leading to the present work as an endeavor to provide a generalized integrated solution towards the management of genomic and clinical data for clinical genomic research.

This PhD thesis represents the description of AD2BioDB, biological data management platform and SeqPipe, dynamic pipeline management software, in the path of meeting the challenges posed in the area of clinical genomics. AD2BioDB provides the platform where data generated using different technologies can be managed and analyzed with reporting and visualization modules for improved understanding of the results among all research collaborators. AD2BioDB is the management software environment in which the in-silico data can be shared and analyzed. The analysis software is connected within AD2BioDB through the plug-in system. SeqPipe software provides opportunity to dynamically create pipeline workflows for the multi-step analysis of data. The interactive graphical user interface provides the opportunity for coding free pipeline creation and analysis. This tool is especially useful in the dynamic NGS analysis, where multiple tools

with different versions are in use. SeqPipe can be used as independent software or as a plug-in analysis tool within an application like AD2BioDB.

The key features of AD2BioDB can be summarized as:

- *Clinical genomics data management*
- *Project management*
- *Data security*
- *Dynamic creation of graphical representation.*
- *Distributed workflow analysis*
- *Reporting and alert features.*
- *Dynamic integration of high throughput technologies*

We developed AD2BioDB as a prototype in our laboratory for providing support to the increasing genomic data and complexity of analysis. The software aims at providing a continuous research experience with a versatile platform that supports data management, analysis and public knowledge integration. Through the integration of SeqPipe into AD2BioDB, the management system becomes robust in providing a distributed analysis environment.

Sommario

Il progresso tecnologico in Genomica ha spinto la ricerca in una nuova era, in cui i metodi di condurre esperimenti sono stati completamente rinnovati. Supportata dall'Information Technology, dotata di strumenti statistici, la Genomica è riuscita a rivoluzionare il campo della biologia fornendo una prospettiva d'analisi impensabile in passato. Con il completamento del progetto genoma umano, è ora disponibile una sequenza di riferimento comune sulla quale poter condurre analisi sull'intero genoma. Ad oggi la ricerca è spinta da una serie di tecnologie d'analisi ad ampio spettro come il sequenziamento, la genotipizzazione e gli studi di espressione genica. In questi ultimi anni, in campo clinico, si sta tentando di integrare in un formato strutturato queste tecnologie. I clinici stanno sempre più interessandosi alla genomica per stabilire, ad esempio, la predisposizione di un paziente allo sviluppo di una certa patologia, permettendo così di applicare la cosiddetta medicina personalizzata per il miglioramento della condizione di salute. Il continuo decremento dei costi e dei tempi di sequenziamento del genoma, nonchè l'incremento dei dati provenienti dalle altre tecnologie ad ampio spettro, ha condotto sia ad un aumento considerevole dei dati genomici pubblici a disposizione della comunità scientifica sia ad un incremento dei requisiti di spazio d'archiviazione di tali dati. L'incremento del numero delle tecnologie presenti nei laboratori d'analisi, così come l'incremento dei tipi di formato dati prodotti da esse, ha condotto ad una maggior necessità di automatizzare la gestione dei dati e delle analisi stesse. In questo quadro i software di gestione di laboratorio (LIMS) stanno diventando il fattore limitante. A ragion di ciò la genomica sta ora cercando di trovare delle soluzioni per la gestione di dati così eterogenei non limitandosi solo alla sfera dei laboratori di ricerca, ma cercando di estendere tale gestione ai singoli medici nonchè alle istituzioni sanitarie. Inoltre, vi è una crescente necessità di integrare nelle piattaforme di gestione ed analisi dei dati delle soluzioni di visualizzazione in grado di garantire un'esperienza di ricerca continua ed una supervisione dell'intero sistema. La sicurezza dei dati è di primaria importanza, in particolare nel settore sanitario, dove si trattano dati sensibili. L'interesse dei medici aggiunge un altro requisito di gestione, un sistema di trasmissione per il soggetto in questione.

L'ipertensione è una patologia complessa diffusa in tutto il mondo. Il progetto HYPERGENES è nato con l'obiettivo di integrare dati e processi biologici utilizzando l'ipertensione come modello di malattia. Il progetto HYPERGENES si focalizza sulla definizione di un modello genetico ed epidemiologico dei caratteri complessi come l'ipertensione essenziale (EH) e fenotipi intermedi dell' ipertensione come il danno d'organo (TOD). Nel corso del progetto HYPERGENES, sono state prese in considerazione e valutate le esigenze di cui sopra. Ciò ha portato al presente lavoro come un tentativo di fornire una soluzione generalizzata ed integrata per la gestione dei dati clinici e genomici per la ricerca clinico-genomica.

Questa tesi di dottorato rappresenta la descrizione di AD2BioDB, piattaforma di gestione di dati biologici e SeqPipe, software di gestione di una pipeline dinamica, nel constesto delle sfide poste nel settore della genomica clinica. AD2BioDB fornisce una piattaforma in cui i dati generati utilizzando diverse tecnologie

possono essere gestiti e analizzati con report e moduli di visualizzazione per una migliore comprensione dei risultati tra tutti i collaboratori di ricerca. AD2BioDB è l'ambiente software di gestione in cui i dati in-silico possono essere condivisi e analizzati. Il software di analisi è collegato all'interno AD2BioDB attraverso un sistema di plug-ins. Il software SeqPipe fornisce la possibilità di creare in modo dinamico i flussi di lavoro di pipeline multisteps per l'analisi dei dati. L'interfaccia grafica interattiva offre all'utente l'opportunità di creare, configurare l'analisi da condurre. Questo strumento è particolarmente utile per l'analisi dinamica di dati di next generation seguencing in cui vengono utilizzati svariati software spesso soggetti a continui aggiornamenti. SeqPipe può essere utilizzato come software indipendente o come plug-in di analisi all'interno di un'applicazione come AD2BioDB.

Le caratteristiche principali di AD2BioDB possono essere così riassunti:

- *Gestione dei dati clinici genomici.*
- *Gestione del progetto*
- *Sicurezza dei dati*
- *Creazione dinamica di una rappresentazione grafica.*
- *Analisi del flusso di lavoro distribuita*
- *Creazione di relazioni e funzioni di allarme.*
- *L'integrazione dinamica di tecnologie ad ampio spettro.*

Abbiamo sviluppato AD2BioDB come prototipo nel nostro laboratorio per fornire supporto all'incremento della quantità dei dati genomici prodotti e alla crescente complessità delle analisi effettuate. Il software mira a fornire un'esperienza di ricerca continua attraverso l'utilizzo di una piattaforma versatile e dinamica che supporta la gestione dei dati, l'analisi e l'integrazione di pubblico. Attraverso l'integrazione di SeqPipe in AD2BioDB, il sistema di gestione diventa solido nel fornire un ambiente d'analisi distribuito.

Acknowledgement

It is not enough just to learn and know, but you ought to possess and own it
- Aristotle

The road of PhD thesis is trodden by many, still each journey is unique by itself. The scientific achievements are pleasant sights along the path, but the individuality of the journey is augmented primarily by the contribution of people who contributes to a PhD student's life. They contribute in paving the path afresh. I feel that without an appreciation of gratitude towards their contribution, this work is never complete.

I thank my parents, my brother and all family members for all selfless efforts and support that they have blessed me with. Life is never easy, but with a considerate and understanding wife, the enthusiasm of facing challenges never diminishes. I thank my wife Shilpi for being the way she is, considerate and caring.

A tutor is rightly called a guide, as he guides you through the unforeseen and unexpected. I express my gratitude to Prof. Daniele Cusi, my tutor and Dr. Cristina Barlassina for their support and guidance. Their mentoring has helped me in understanding many aspects of science. Further, a student's life is molded by the colleagues and fellow workers. For someone starting life in another country with a new language requires a lot of efforts and I thank Eng. Maurizio Mercurio, Dr. Erika Salvi and Eng. Leopoldo Fratti for their concern and help. I will never forget the day when I went for my first apartment and the owner of the house and I could not interact in a common language. Our entire conversation was mediated by Maurizio, who translated our sentences over a telephonic conversation. I am grateful to Dr. Sara Lupoli for painstakingly revising the thesis and her multiple advises. I am indebted to Dr. Matteo Barcella for helping me translate the abstract into Italian. I also thank all my colleagues for their support. The coffee sessions with Daniele Braga, Dinesh and Matteo were a welcome break between works. Further I am also grateful to Eng. Andrea Calabria for introducing me to the ADempiere software.

I feel bad for the missed moments with my friends, in India. Nonetheless, I am grateful for having you in my life and look forward to the time when we can be together and have fun. I appreciate the new friends that have come in my life during the last years. Their contributions have been immense in riding over the rough patches of the journey.

List of Abbreviations

LIMS	Laboratory Information Management System
EH	Essential Hypertension
TOD	Target Organ Damage
HGP	Human Genome Project
DNA	Deoxyribonucleic acid
NGS	Next Generation Sequencing
WHO	World Health Organization
ICD	International Coding of Diseases
SNOMED	Systematized Nomenclature of Medicine
UK	United Kingdom
CT	Clinical Terms
LOINC	Logical observation identifiers names and codes
HL7	Health Level Seven
DICOM	Digital Imaging and Communications in Medicine
CEN	European Standardization Body
EHR	Electronic Health Record
ANS	American National Standards Institute
CIDSC	Clinical Data Interchange Standards Consortium
SDTM	Study Data Tabulation Model
SRA	Sequence Read Archive
XML	Extensible Markup Language
INSDC	International Nucleotide Sequence Database Collaboration
NCBI	National Center for Biotechnology Information
EBI	European Bioinformatics Institute
DDBJ	DNA Data Bank of Japan
IMG	Integrated Microbial Genomes
TCGA	The Cancer Genome Atlas
SRF	Sequence Retrieval File
BWA	Burrows-Wheeler Aligner

GATK	Genome Analysis Tool Kit
SAM	Sequence Alignment Map
BAM	Binary Alignment Map
CPU	Central Processing Unit
SNP	Single Nucleotide Polymorphs
UCSC	University of California, Santa Cruz
DBMS	database management system
SQL	structured query language
DTL	data transactional language
ACID	Atomicity Consistency Isolation Durability
TCP	Transmission control protocol
HTTP	Hypertext Transfer Protocol
FTP	File Transfer Protocol
POP3	Post Office Protocol
SOAP	Simple Object Access Protocol
SSH/ SFTP	Secure Shell File Transfer Protocol
MVC	Model View Controller
ERP	Enterprise Resource Planning
GPL	GNU Public License
UI	User Interface
API	Application programming interface
GWA	Genome wide association study
PCR	Polymerase Chain Reaction

List of Figures

List of Tables

Table of Contents

1 Introduction

Information Technology has completely transformed the way biological research is carried out. Laboratory automation and new technologies have transformed experiments. In this era of the “New Biology” [1] data generation time is decreasing and the data volume is enormously increasing, where biology is becoming industrialized. High Throughput methods have paved the way for advances in knowledge with a speed, unimaginable in the past. The Human Genome Project (HGP) [2] pushed the frontiers of the quest of knowing ourselves to higher realms. For the first time large population based studies can be performed for universal understanding of species. The effect of technology can be seen with the new disciplines, new terminologies in biology differentiated from traditional themes markedly due to the technological perspective. We are in the “omics” era [3] where computational analysis of complex experimental protocols is a daily norm. Omics suffix signifies the totalitarian approach to evaluation at a given biological level. Automation of experiments is thus fast increasing to perform multiple repetitive tasks. Single analysis steps are giving way to multistep workflows for a broader understanding of the experimental study. The knowledge of the entire human genome finds its utility not only in biological research but extends to the healthcare. With genome wide research becoming more affordable, interest of health care sector is increasing for the utilization of omics technologies for providing better care to patients. Clinicians are attracted towards the use of genomics finding genetic cause for complex diseases and for providing personalized medical care. As a controlled group patients respond to the same therapy differently, clinicians appreciate the need for identification of a measure of the patient’s profile. This profile can be achieved through genomics and thus the screening of patients for particular therapies can be followed.

But, all these technological advances have come with a price! The use of High Throughput methods have resulted in a data deluge that has to be channeled appropriately [4]. Infrastructural needs have increased for the storage of the generated data. Biological data requires proper storage with fast retrieval; with most of the data electronic in nature, scientists have been turning to sophisticated computational methods. Also the analysis tools are being re-evaluated under this new challenge. The need for data sharing is becoming a driving force towards standardized formats. Under these requirements, in the field of biological research the management of data has become a daunting task. The development of fields of genomics and health care have been guided by diverse objectives and rules; with the present interest of clinicians in genomic technologies implies the redefinition of the practices and procedures in an interdisciplinary environment of collaborative work.

Automation of the research laboratories had begun with the Laboratory Information Management System (LIMS) that manages laboratory data. These systems included the sample management, instrument and application management, electronic data exchange, barcode handling and basic workflows. The present data volume requires the automation process be extended far beyond the basic

information management to advanced management systems that carries LIMS functionality as well as has enhanced analysis capabilities with distributed reporting of the results. These software needs to be flexible to the technological changes and user friendly with easy accessibility. Interoperability of formats is the basis for such a flexible system to function and needs to be in accordance with the defined standards.

In this present scenario the need for a system, dynamic enough to incorporate the changing requirements and be robust to manage the distributed infrastructural environment, is felt by genomics scientists as well as clinicians. There is a growing need for a clinical genomic data management system that can incorporate functionalities of LIMS and can be scaled to dynamic platform for analysis and visualization. The system should be able to assimilate heterogeneous data and be able to carry out analysis of varying complexities. Visualization of genomics data has always been missing although graphical representation provides greater information and comprehension. Apart from visualization, there is the public annotation knowledge available in separate data repositories. This knowledge is essential for proper understanding of the experimental findings and for the identification of research directions.

1.1 Description of chapters

Chapter 1: Describes the new disciplines that have come to prominence using the technological advancements. Then is introduced the present problem related to management of voluminous high throughput data in a dynamic environment of technological evolution. It is also discussed the main challenges related to the Big data.

Chapter 2: Description of the features essential in the present scenario for a clinical genomic data management system and workflow management.

Chapter 3: Identifies the technological standards relevant for the development of dynamic and adaptable software that attempts to provide a complete research experience in a distributed environment.

Chapter 4: Explains the implementation and the development path of the software.

Chapter 5: Describes the configuration details for the sequencing technology. It also describes a case study using sequencing data from the HYPERGENES project.

Chapter 6: Concluding remarks about AD2BioDB and SeqPipe and discusses about the advantages this prototype system provides in the clinical genomic domain.

Chapter 7: Identifies the present limitations and also gives us a glimpse of the future development possibilities and identifies some of the features that can be integrated according to the requirements identified by domain specific users.

1.2 Background

1.2.1 Genomics and high throughput methods

***"Genomics** is defined as the study of genes and their functions and related techniques."*

- Who definition [5]

The word *Genomics* was first coined in 1986 by Dr. Thomas H. Roderick, a geneticist at the Jackson Laboratory, Bar Harbor, ME. Genomics, the study of organisms' entire genomes, includes intensive efforts to determine the entire DNA sequence of organisms and fine-scale genetic mapping efforts. The progress of genomics is summarized by Brosnahan, Brooks and Antczak [6] as

"*The term "genomics" has supplanted that of "genetics" as research focus has shifted from single genes and their protein products to considerations of how the products of multiple genes interact to produce complex traits and how genes are regulated. The impact of genomic study is far-reaching, encompassing not only the obvious identification of specific disease-causing mutations but also expanded knowledge of normal physiology and insights into the evolution (…)*"

Frederick Sanger's di-deoxy terminator sequencing technique led to the establishment of Genomics and has been backbone technology for DNA sequencing for the last 40 years. Over the past years, improvements in the sequencing technologies are restructuring genomics research. These high-throughput methods have reduced the sequencing costs making the technology affordable and widely available for research. High throughput methods describe the parallelized use of resources over period of time to accomplish a given task. These methods can carry multiple experiments in an automated environment and achieve high performance. Common sequencing applications include whole or target genome sequencing, variant detection, gene expression profiling.

Alternative sequencing strategies like pyrosequencing ([7], [8]), reversible terminator chemistry [9], sequencing-by-ligation [10], virtual terminator chemistry [11], real-time sequencing [12], microfabricated high-density picolitre reactors[15] and whole-genome re-sequencing[16] have reduced time 100 to 1,000 times and cost of sequencing to 4%-0.1% of Sanger sequencing. These strategies taken together are represented as the Next Generation Sequencing (NGS) [13]. Kircher and Kelso [14] discussed some of these methods in details.

Microarray technology is another high-throughput technology that has resulted in large scale experiments. This versatile technology is used for parallel gene expression analysis for thousands of genes of known and unknown function, or DNA homology analysis for detecting polymorphisms and mutations. This technology is based on the hybridization of static probes to the complementary DNA strand of the sample. Maksos and Southern [17] first demonstrated the synthesis of arrays of oligonucleotides on a solid support in situ. Advances in

technology and chemistry resulted in increasingly higher density oligonucleotide microarrays synthesized in situ using techniques such as photolithography [18] and ink-jet deposition [19]. Widespread adoption of microarray technology in both industry and many academic research laboratories was mainly because it quickly and accurately performed simultaneous analysis of thousands of genes in a massively parallel experiment, providing extensive and valuable information about the gene function. The main limitation of this technology is the requirement of sequence knowledge. Microarray technology diminished in importance in the light of the genome sequencing technologies but still has significant relevance for the research, as described by Gresham, Dunham and Botstein [20]:

"The applications that are envisioned for cheap and rapid sequencing technology do not actually require repeated determination of entire genomic sequences. Methods that efficiently detect genomic differences be they structural rearrangements, polymorphisms or mutations, often suffice to reduce the sequencing requirement to a tiny fraction of the genome, a capability that is routine in most modern biology laboratories. Several technologies that use hybridization to DNA microarrays are effective for detecting genomic variation in closely related samples. Thus, questions in which a researcher aims to compare normal and diseased tissues from the same individual or mutant and wild-type DNA from the same experimental organism can often be addressed by microarray-based experimental comparison as opposed to exhaustive sequencing of entire genomes."

Although sequencing technology is becoming cheaper the associated infrastructure required for regular use in research is still expensive. Thus the array technology has its relevance in many laboratories. Gresham, Dunham and Botstein also highlight the potential of microarrays to provide insights into human genome variation ([20]– box 3).

1.2.2 Clinical genomics and personalized medicine

Clinical Genomics can be defined as the use of genomic technologies for medical purposes. The clinical genomics domain addresses requirements for the interrelation of clinical and genomic data at the individual level [21]. In the past many diseases have been understood in the light of the molecular biology and genetics, but the large number of complex diseases and new lifestyle diseases are left unanswered. "Complex diseases are caused by a combination of genetic, environmental, and lifestyle factors, most of which have not yet been identified. The vast majority of diseases fall into this category including several congenital defects and a number of adult-onset diseases." [22]. Clinical researchers working for understanding the molecular aspects of these diseases are increasingly drawn by the recent advances of genomics. Genomics provide clinicians to better understand the human genome through large population based studies. Who highlights the benefits of the genomic knowledge in the following main categories [62]:

- Clinical diagnostics and predictive testing.
- Identifying new treatment.

- Developing preventive measures.
- Direct economic benefits

In diagnostics the genomic profile of the patient identifies disease markers thus helping disease association and can also be used for the prediction of disease. For example, genes that enhance susceptibility to Type 1A Diabetes have been identified and can be used to assess disease risk [23]. In this case, preventive care can follow for patients with greater risk. Genomics also provides a platform for comparison of a person's genome with healthy individuals thus identifying the unique and novel changes in marker profiles leading to new treatment protocols. Combining clinical and environmental data with genomic data enables more efficient and accurate identification of targeted gene expression and the effect on treatment results.

In the past, the tools of genetics have been restricted for a clinical setting, but the HGP has led to the development of substantial new technologies that are capable of defining large sets of biomarkers systematically in biological samples. These new approaches to monitoring disease through genomics have applications in clinical diagnostics and predictive testing. Knowledge of predisposition or disease sub-types would lead to earlier monitoring or treatment. For diseases with multiple sub-types a diagnostic test can be critical in choosing the appropriate treatment. Gene-based testing is expected to provide better targeting i.e. whom to intervene and tailoring, the best method for intervention and preventive efforts. Never the less, the advantages of population-wide vs. high-risk preventive strategies also depend on disease prevalence and the role of other major risk factors. The genomic targets are mainly used in pharmaceutical studies for the increase of drug efficacy and for new drug development. Variant forms of drug responses can also be assessed in the light of the genomic evaluation. Presently, the main efforts of genomics in the clinical domain are directed towards the understanding of the molecular basis and genomics of complex diseases. John Bell described the use of genomics as

"*Molecular phenotyping using genetic and genomic information will allow early and more accurate prediction and diagnosis of disease and of disease progression. Medicine will become oriented towards disease prevention rather than efforts to cure people at late stages of illness." [24]*

"Variability is the law of life and as no two faces are the same, so no two bodies are alike, and no two individuals react alike and behave alike under the abnormal conditions we know as disease".

- Sir William Osler (1849-1919)

The applications of clinical genomics are achieved through the health care institutions. Personalized care towards patients has been provided through an interactive process by the physicians based on the disease symptoms. The medications are specific to the disease condition. It is a constant effort towards betterment and the use of new knowledge, insights and technology in this path has been consistent. Information technology now tends to redefine the way in which

care is delivered. Rapid availability of information and automation of testing laboratory are providing swift care to patients. Increase of genomic knowledge and significant reduction in the cost of sequencing now provides the opportunity of being used in health care. The term 'Genomic medicine' is now being used interchangeably with the personalized medicine signifying a medical system with care tailored according to the genomic individuality. With these varied changes the health care informatics tries to formulate a functional system providing great impetus to personalized care.

1.2.3 Health care and standards

The health care informatics is a broader term that encompasses clinical informatics. While the clinical informatics is targeted towards the use of technology for the clinics and research; health informatics covers areas of administration of the system including all aspects, from billing to treatment at the point of care [25]. Interoperability is the pivotal factor on which modern health care informatics is being formulated providing a continuity of information across collaborators, shedding infrastructural barriers. The first major step is the gradual and focused shift towards the universal, standardized Electronic health record [26]. This record system takes into account the ethical and socio-legal issues in keeping the personal information secure, within the distributed framework. The transition from the paper based system to the electronic format has many challenges that are gradually being answered. Care for patients is provided in evidence based standardized system for having a better effect and being secure. The role of standards in the health care system is eminent and essential. Incorporation of genomic knowledge into clinical practice, involves the designing of a standardized model. In this regard the present focus is on the protection of privacy of the patients, the data sharing rules and the use of knowledge gathered from the usage of the data. Privacy is maintained by the secure coding and screening of all information that relates to an individual. Data sharing with other associated features gains emphasis with the increase of data in the public domain through large Bio-banks and other large data repositories. Wolf et. al. discusses the concerns about the use of large archives for the storage of data and also identifies 10 recommendations for implementation of archives [27]. There are many existing standards being used in the health care system. The notable ones include the International Coding of Diseases (ICD) [28] maintained by the World Health Organization (WHO) for reporting, Systematized Nomenclature of Medicine (SNOMED) [29] is mandatory in the new program for health information technology in UK. The SNOMED Clinical Terms (CT) core terminology contains healthcare concepts with hierarchical organization and unique meanings and formal logic-based definitions. Logical observation identifiers names and codes (LOINC) [30] is a terminology that facilitates the exchange and pooling of testing results used in clinical care, outcomes management and research. Most laboratories and other diagnostic services use Health Level Seven (HL7) [31] messages for data exchange. HL7 is an ANSI-accredited standards developing organization in healthcare. Interoperability among units of the care delivery system can only be achieved by the use of universal coding system, recognized and implemented universally. Digital Imaging and Communications in Medicine (DICOM) [32] is used

for the exchange of medical imaging data. European Standardization Body (CEN) [33] offers a viable EHR standard. Other data reporting standards includes the ISA-Tab [34], MAGE-Tab [35], Clinical Data Interchange Standards Consortium (CDISC) [36] data model, the Study Data Tabulation Model (SDTM) [37], Sequence Read Archive (SRA) schemas governing XML metadata (SRA-XML) [38] developed under the International Nucleotide Sequence Database Collaboration (INSDC) [39]. Similar standards exist for the genomic data, but with respect to the clinical domain require modifications. A detailed discussion about standards is covered later.

Hudson L.K. reviews [40] the integration challenges of genomics in the clinical domain and identifies four major areas of concern- Consent and Confidentiality, Return of Research Results, Regulation of Genetic Tests, Law Enforcement. While first two are discussed above, the legal aspects are being formulated in different countries at various levels of adoptions. One of the major concerns relates to genomic discrimination among individuals and extends to the use of genomic information by insurance companies. Ellen Wright Clayton summarizes these in [41]. The present developments creates certain paradoxes with the fundamental principles for the management of health care system [42], as adaptation vs. stability and continuity; Decentralization, differentiation vs. centralization, integration ; An environment that is error-intolerant but blame free; Democratic open forum for ideas coupled with the autocracy of ultimate decision-making. A suitable solution is needed for the smooth transition into an efficient utilization of the potential of genomic medicine.

1.2.4 Data Management

The improvements in the high throughput technologies are followed by the concern for the management of the data being generated. For the last two decade, technology has reshaped biological research and the growing concern has been expressed all along ([43], [44], Figure 1). The international archives of biological data maintained by National Center for Biotechnology Information (NCBI), European Bioinformatics Institute (EBI) and DNA Data Bank of Japan (DDBJ) now have to maintain a staggering amount of data which runs in terra-bytes.

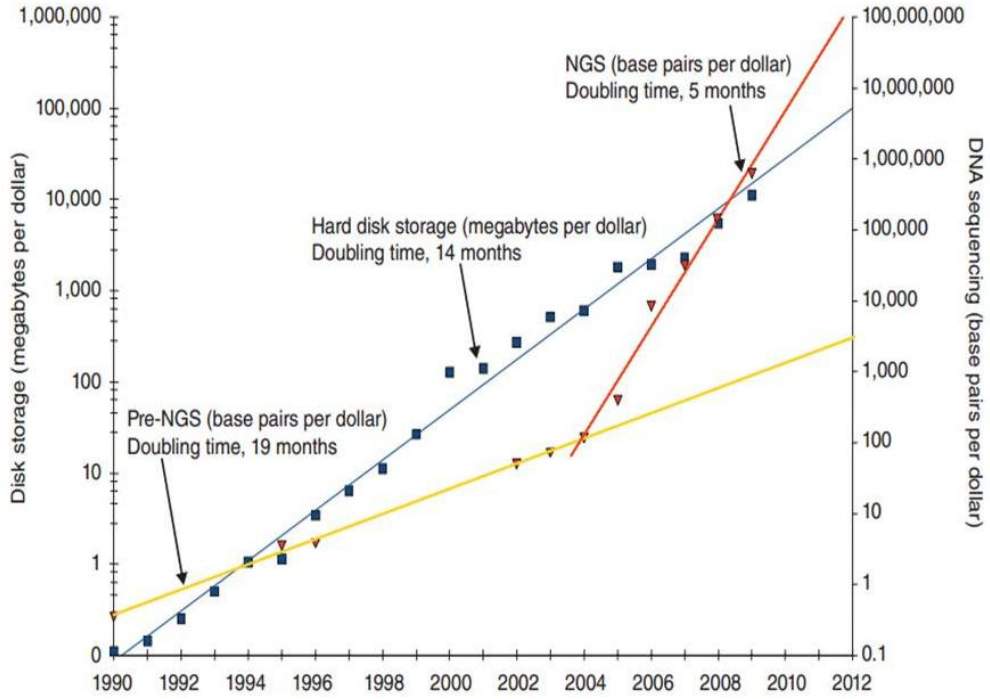


Figure 1 Next generation sequencing and storage [61]

In health care scenario, this size tends to grow exponentially as the genomic technologies are adapted into the mainstream of the care delivery [45]. This is mainly due to routine use of the technology for the understanding of the clinical conditions of the patients. Further the life time maintenance of patients' data increases this volume even further. The growth of storage hardware compared to the present needs is of great concern and as an alternative solution grid based and cloud based distributed systems are being used. In the clinical domain these distributed systems generated concerns for data security. Large scale data management involves factors of interoperability, controlled access, dynamic query system for quick data retrieval. Adaptability of the management system is another point of emphasis as newer technologies like sequencing are rapidly developing and there is always the possibility of novel technology being adopted in the future. This adaptability is reflected in the underlying data schema or the format in which the data is stored. On the other hand all data must be analyzed to mine for meaningful information. With the large sized data manual transfer of data for the analysis programs becomes cumbersome and inefficient. It is advantageous and more convenient to link the analysis applications to the data repository. This association requires the homogeneity in the transformation of the data formats and is a cause of concern as novel technologies tends to have many practiced formats with standards being defined further in the development cycle. Thus automation of process becomes complex. Further the results of analysis need to be homogeneous for use in multiple domains. This identifies a close relation between

the data store planning and the data (re)usability. Meaningful generalizations of data are essential, yet user specific querying must be efficient in returning true results with the least false positives. Biological data is primarily stored in files while most developments for databases have been directed towards the relational databases, which provide greater data management. Thus the storage system needs to be addressed properly.

Sansone et al [46] provides a great insight into the requirements for interoperability in biological data. The potent role of an interoperable system is in data reusability and reproducibility of research and extends to the reporting system for result sharing. Complex experiments involving multiple technologies are becoming common through the use of workflows and pipelines, thereby increasing the interoperability process. Interoperability is dependent on the standardization of data formats for ease of access. In the clinical domain, interoperability is the basis of further developments i.e. the decision support system. In this domain data security issues also needs to be accounted for. Reporting systems for clinical genomics also need to account for better visualization strategies, comprehension and representation of complex analysis. Genomics data is mainly visualized with the help of genome browsers and their use in the clinical domain requires many rules for control of data access.

1.3 Genomic Data Management

Data management needs for genomic research has developed with the high throughput technologies and the voluminous data generated but, in the clinical domain, data management systems have evolved to maintain patient data with standardized operation protocols and data security standards. The main parts of a clinical data management system include –

- Database design for the logical categorization of the data. This feature is the basis for a flexible and adaptive system.
- Access control and monitoring of users.
- Data entry and screening for quality maintenance.
- Discrepancy management and fast query resolution.
- Security of studies and data.
- Extraction of data for reporting and analysis.
- Account and access management.

Clinical genomic data management requires all the above mentioned features but extends to specialized data handling capabilities. The technological advancements in the genomic field emphasize the need for a flexible and dynamic system that can be improved with minimum programmatic efforts. Visualization of genomic data is complex requiring specialized browsers and sometimes browser specific data formatting. Further the capacity to handle multiple studies with voluminous data is essential. Genomic knowledge is distributed over many repositories and hence access to these databases with intelligent query capability becomes essential for proper analysis.

Two major initiatives for the inclusion of research data for the clinical studies include a) the Informatics for Integrating Biology and the Bedside (I2B2) [47], which aims to build an informatics framework that will bridge clinical research data and the vast data banks arising from basic science research, thus providing a better understanding of the genetic bases of complex diseases and b) Cancer Biomedical Informatics Grid (caBIG) [48], project intended to develop a collaborative information infrastructure that links data and analytic resources within and across institutions connected to the cancer grid. These systems are a comprehensive step towards the incorporation of genomic data in the clinical domain. I2B2 is developed following a pluggable framework consisting of hives. Genephony [49], GenePING [50], iRODS [51], Integrated Microbial Genomes (IMG) data management system [52] software provides management capabilities for the genomic research laboratories. Further the LIMS provide data entry and basic management support to the high throughput genomics laboratories.

1.4 Genome Sequencing experiments- pipelines and workflows

Genome sequencing technology stands behind the giant leap of genomics. The post genomic era, marked by the completion of the HGP, saw a major improvement in the sequencing capabilities with reduction in cost and time and the beginning of 1000 genomes project [53], The Cancer Genome Atlas (TCGA) [54] and the 10,000 genomes project [55]. The major use of the sequencing technology includes de novo sequencing or whole genome sequencing, SNP calling, Structural variation analysis, Methylation analysis and RNA sequencing. Analysis conducted using sequencing involves whole genome sequencing to characterize variants across the entire genome, targeted gene sequencing for selection of “targeted” genes across many samples, structural variation by sequencing mate-pair library at fairly low coverage of the genome (variations are detected with genome-wide coverage) and transcriptome characterization for assessment of transcript abundance and/or splice variants. The stages in the next-generation sequencing process involve:

1. Base-calling is the initial analysis carried as a part of the vendor supplied sequencing process. It involves the identification of particular reactions. The results of this step include the identified base, quality score and intensity values. SRF file format is preferred but vendor specific formats are also present.
2. Filtering is mainly carried according to the type of analysis being carried out. However some filtering is done during the base calling step for screening of artifacts of the sequencer and for improving accuracy. Filtering is mainly done for paired reads and duplicate elimination.
3. Mapping is the alignment of the reads to a reference sequence. The selection of the reference depends on the type of study conducted in order to avoid reference bias.
4. Alignment is carried out mainly when there is no reference sequence in order to assemble sequence reads into contigs and scaffolds.

Development of algorithms for alignment is an active area of research in order to achieve more accurate and fast methods.

The output files, after previous step 3 or 4 contains the sequence and information about the match. Burrows-Wheeler Aligner (BWA) [56] and Genome Analysis Tool Kit (GATK) [57] are currently the most popular tools for raw sequence data alignment and variation calling respectively. This step is sometimes followed by another filtering step to eliminate non unique matches, matches for repetitive regions etc.

5. Downstream analysis follows, depending upon the purpose of study. This area is the most developing part as new studies are being formulated for the use of sequencing technologies.

The common file formats involved in sequencing experiments include:

1. SRF and Fastq formats for raw read data. The SRF format is binary and highly compressed, it contains sequence reads, quality values, and intensities from the entire sequencing run and so can still be large. The most common formats used in next-generation sequencing are the FASTQ format for raw reads, first used on data from the Illumina/GAIIx platform.
2. SAM/BAM (text/binary versions) [58] format for post alignment data. BAM file stores only the information of the primary read's alignment to the reference sequence in a compact form that is more valuable than raw sequence data. Also only the reads mapped/aligned to genome is retained thereby maintaining quality. It is platform and tool independent. BAM format is widely used but does not allow for rich metadata, in fact only a few header fields are available.

The sequencing experiments require multi-step analysis which is further complicated by some extensive downstream analysis. The sequencing workflows in turn are dynamic systems that have to be flexible to adapt to changing technology, and growing tools with greater accuracy. Further, the workflow management requires extensive computational power for analysis and storage. In the distributed environment, the data transfer capabilities through the network needs to be fast for continuity of development and efficient analysis.

Genome centers are now turning to the cloud based technologies to carry large scale analysis and running of the workflows. This approach involves setting up a local cloud system enabling higher speed of data transfer with the significant reduction in analysis time. The alternative approach to the huge investment is the renting of cloud time resulting in reduces the cost of infrastructure and maintenance to a cost of renting. The long term cost effectiveness of regular use of this approach will be seen in future. For sequencing laboratories a distributed sequencing workflow management system is better suited to link available resources for analysis. Present solutions include Galaxy [59] is a web-based workflow system, preconfigured with many next-generation sequencing tools, Taverna system (http://www.taverna.org.uk/) [60] particularly if the analytical services are web-based rather than strictly local.

1.5 Automation challenges with high throughput genomic data

Whenever we talk of genomics experiments the primary challenge is size. Managing large and voluminous data has been challenging. Previously, processor improvements and parallelization of processing cores provided the power to cope with the increase in volume. These developments followed Moor's law for comprehension. Now the data size is outpacing the development of computational resources and has lead to massively parallel clusters to manage data. Distributed systems are trying to cope with these needs. Storage media has also developed from hard disks to solid state devices to store and provide data at a greater speed. The hurdles related to infrastructure are heavily inspiring the development of computer hardware.

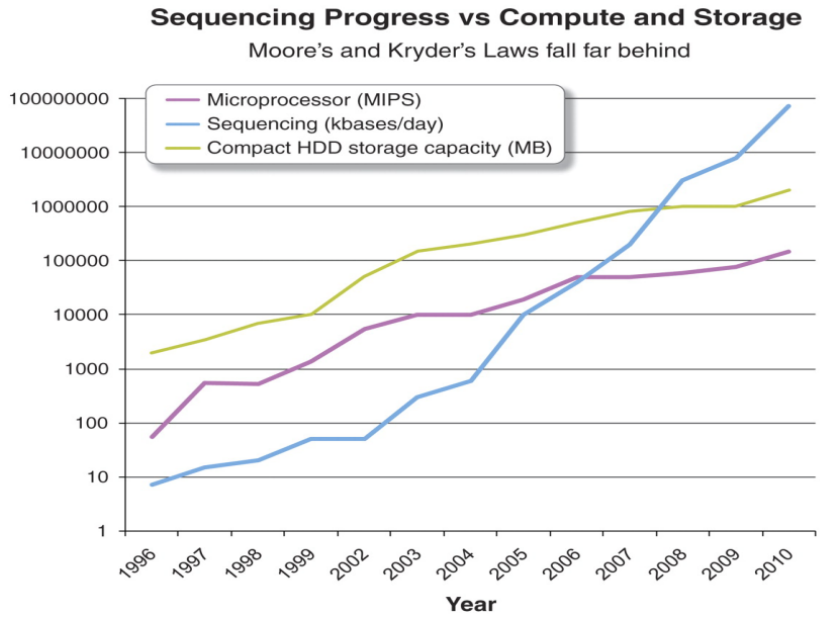


Figure 2 Sequencing data generation driving computational hardware development

The main challenges high throughput genomics management and automation is discussed here.

1.5.1 Storage and archiving

The primary concern for any genomics lab is storing the high throughput data. Data volume has led to the use of data clusters. With these high capacity infrastructures, still genomics laboratories are unable to store the primary data generated from the sequencing experiments. The primary data are stored temporarily until the secondary analysis is completed and subsequently the BAM files are archived. The tertiary analysis further adds to the data volume at latter steps. Further high

throughput data is file based and cannot be directly integrated into any relational database. Flat file databases have many drawbacks and hence relational databases are used for storage of information related to data while the actual data is linked through the physical path to the data file. This hybrid scenario does provide some relational flexibility for the metadata but the experimental data faces the limitations of the flat file databases. It is harder to interchange data formats and complex querying is ineffective within the data files. These limitations highlight the importance of a highly descriptive metadata for the stored data such that the archived data can be efficiently used.

1.5.2 Integration

High throughput data technologies are diverse and so are their respective data types. Integration of these data for the generation of knowledge poses a great challenge. The varying degree of metadata makes integrated mining a very difficult task involving a lot of human effort. This has become a serious bottleneck in the automation process. While in microarray technology, having reached a greater maturity, it has been defined a minimum set of metadata descriptors for universal implementation, with sequencing technology a consensus is yet to be reached. Sequencing technology is fast evolving and consequently also its data formats. Great importance is being advocated for the use of descriptive metadata but a minimal set of descriptors needs to be achieved.

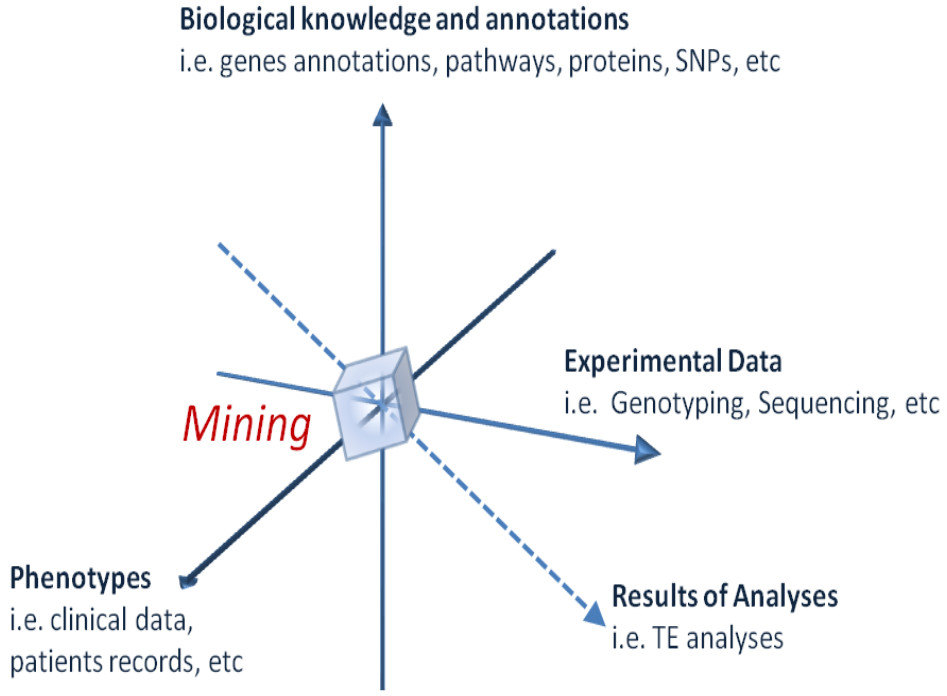


Figure 3 Dimensions of high throughput data mining. The experimental data can be combined with clinical data and the publicly available annotation information for a better analysis.

Further, integration of annotation information available in public repositories is an essential component for understanding the results. Adding to the complexity, these public repositories do not have uniform schematics of their layout and terminologies. In fact current researches in ontology and semantics are directed towards this.

1.5.3 Query Processing

Querying is the basis for the data retrieval, slow response of the system in returning query results impedes overall functioning. The inherent file based genomic data makes querying difficult and slow. Further, the distributed nature of the public annotation knowledge makes simple queries slow. Complex querying to gather useful information in the present mesh of data repositories is thus even more challenging. Complex queries involve multiple data elements and publicly available annotation information from different repositories. For an interactive system the query builder is an efficient feature providing the user the opportunity to create queries dynamically, tailored to the user's needs. This feature can only function efficiently if the underlying system can handle complex queries efficiently.

1.5.4 Interpretation

Genomic data is interpreted through the use of publicly available annotation knowledge and statistical analysis. The most commonly used tool for the representation of the genomic information is the genomic browser. Through the genomic browser the annotation information can be compared to the individual subjects under study. Further analysis can follow the observations from the browser. The challenge is to automatically generate the right metadata to describe the data. Metadata acquisition system can automate creation of metadata and be extensive enough to provide data provenance.

1.5.5 Visualization

Visualization tools for genomics data have been tailored for different analysis needs. They provide the ability to display data and biological annotations from divergent sources in a graphical representation. With the presence of large NGS data these tools are posed with architectural as well as representational challenges. Genome browsers are primarily web based and representing the large data formats is a significant challenge. Further the graphical representation of traditional browsers i.e. the view of multiple sequences as alignment provides little help in cases where the number of reads increase significantly. Support to the new data formats is another consideration as the field of genomics is rapidly evolving with a gradual process of standardization of formats. Different tools are presently available but for a visual support to comprehensive analysis covering different areas of genomics, require either the development of a single tool to provide for all needs or existent tools be inter-connectable. Integrating analysis tools with browsers increases the power of fast analysis. The ability to visualize user data along with the public data are extended by the growing need to visualize and

compare private clinical data requiring the use of security protocols for data protection.

1.5.6 System Architecture

Data processing and analysis involves diverse workflows. These may be CPU intensive requiring more computational processing or may be database intensive with the involvement many queries. Thus the underlying system architecture has to be flexible and adaptive to optimizations required for the divergent work profiles. As described earlier standard database optimizations cannot be used directly, but can inspire the development of new techniques. As technology reshapes the future of biological research and the automation of laboratories being realized the need for a dynamic architecture is even more important. The varied requirements of the biological research demands for a system architecture that is scalable for implementation in a desktop environment to a large research facility with distributed environment.

1.6 Objective of the Thesis

This thesis documents the development of AD2BioDB, a management and analysis platform for genomic high throughput data for use in clinical genomics research. AD2BioDB aims to manage in-silico genomic data obtained after primary experiments, to provide a platform for subsequent analysis and reporting. The overall objective of developing AD2BioDB is to provide a dynamic system that manages projects configured to related technologies with minimum programming efforts and carry in-silico experimentations on the same platform. AD2BioDB strives to provide users a continuous research experience with dynamic project management capabilities.

Further, as a demonstration of functionality within AD2BioDB, this work also covers the development of SeqPipe, a dynamic sequencing workflow creation tool and its use within AD2BioDB for sequencing data analysis. SeqPipe targets another important sub area of the entire problem, the dynamics of workflows. The use of multiple tools having become a norm for most experiments, SeqPipe tries to provide a graphical environment for the creation of workflows dynamically with minimal programming and to support a distributed analysis environment.

2 Characteristic requirements

Carrying a complete research on the same management platform can help in achieving complete automation. This can be divided into data management consisting of data related to experiments, public annotations and clinical phenotypes; analysis and visualization. The management of projects is also important in this process. Presently the research laboratories have separate initiatives for each aspect while a manual coordination needs to be carried out for a complete functionality of the research process. A robust and dynamic system is needed for achieving such goal and objectives.

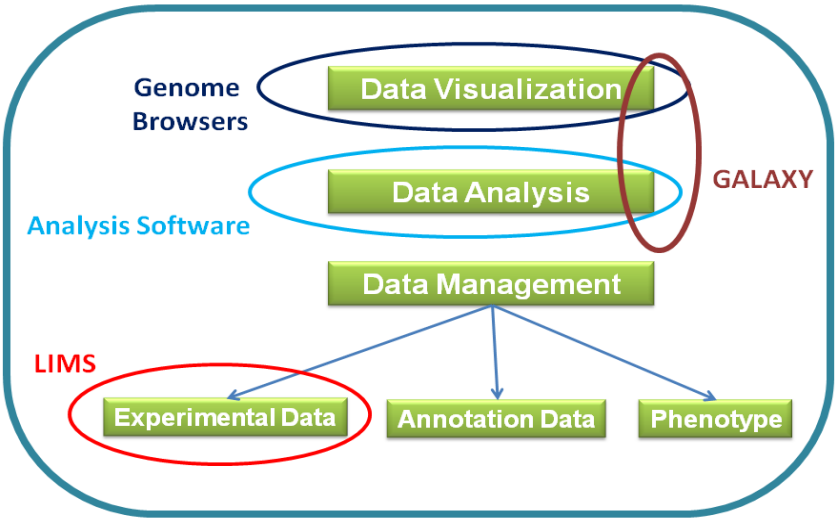


Figure 4 Aspects of genomic research management

The overall objective of realizing a management system can be divided into a number of rational modules. These modules represent independent areas of development covering an important aspect leading to the functioning of the entire system. Within each module a sub domain of development is identified and their status is discussed.

2.1 Clinical genomic data management modules

2.1.1 Archiving and connecting heterogeneous Data

The heterogeneous data in clinical genomics consists of 1) experimental data generated by the use of different technologies, 2) clinical data phenotypes and 3) annotation data represented as public knowledge repositories.
High throughput experimental data depending on the technology used can be sub-divided into:

i) Array technology data related to the genotyping experiments and gene expression. Genotyping experiments identify the presence of particular single nucleotide polymorphs (SNPs) in a targeted region of the genome, while the gene expression studies measures the activity of a particular set of genes. These experiments are carried out in parallel for large number of candidates, thus providing voluminous data for analysis. The data contains intensity values representing the detection of signal.
ii) Sequencing data mainly related to whole genome sequencing, exome sequencing and target re-sequencing experiments. These data contains a portion of the genome as reads and are primarily contained in the SAM/BAM format.

Clinical phenotypes represent clinical observations, physical or biochemical characteristics of an organism determined by both genetic makeup and environmental influences. Phenotypes are described in a hierarchical representation, with each phenotype having a measured value associated with each subject. These observations helps to better understand the experimental data and are used in classification and grouping of study subjects. The first two categories of data can be considered as private as these data are primarily limited to the laboratories involved and are generated following a specific study.

Public annotation data consist of the knowledge associated to particular regions of the genome. These are stored in public databases and are compiled based on scientific evidences. This wealth of knowledge is stored within databases and is available for reference. While UCSC and ensemble annotation databases are most common, a large number of other annotation databases and metabases are also present, some of which are specialized for specific organisms. The large number of repositories available with heterogeneity in representation creates a unique challenge for combining all the available knowledge. Although present data sources provide automatic software mediated access to data through structured queries, but it becomes difficult for dynamic user queries involving multiple data sources due to differences in

i) Query language variation that depends on the database type.
ii) Structural differences in the representation of the database tables for the same topics
iii) Naming differences in the columns and fields
iv) Semantic differences arise when the same concept is represented in different forms like the use of binary value for the occurrence of an event along with the value if the event occurs or a value representation for measuring the event with a 0 indicating the absence of the event.

Heterogeneous database integration system ([63], [64]) provides a global persistent interface for applications representing the data of the same type to be present at the same place. The different kinds of integration methods are-

i) Vertical integration consisting of semantically similar data from different sources.
ii) Horizontal integration composed of semantically dissimilar data sources.

iii) Integration for application portability through the standardization of access to semantically similar information in different sources.

Strategies for providing uniform interface to the heterogeneous databases include

i) Data translation where the data from heterogeneous databases are translated into a local database following a uniform schema. The main drawbacks are a) the data redundancy and infrastructural needs b) modifying the data layout to attain a common layout may lead to loss of information.

ii) Global data model being maintained in the integration system as a minimalistic and abstract model. Abstract semantic data models can provide the common representation for different databases implemented.

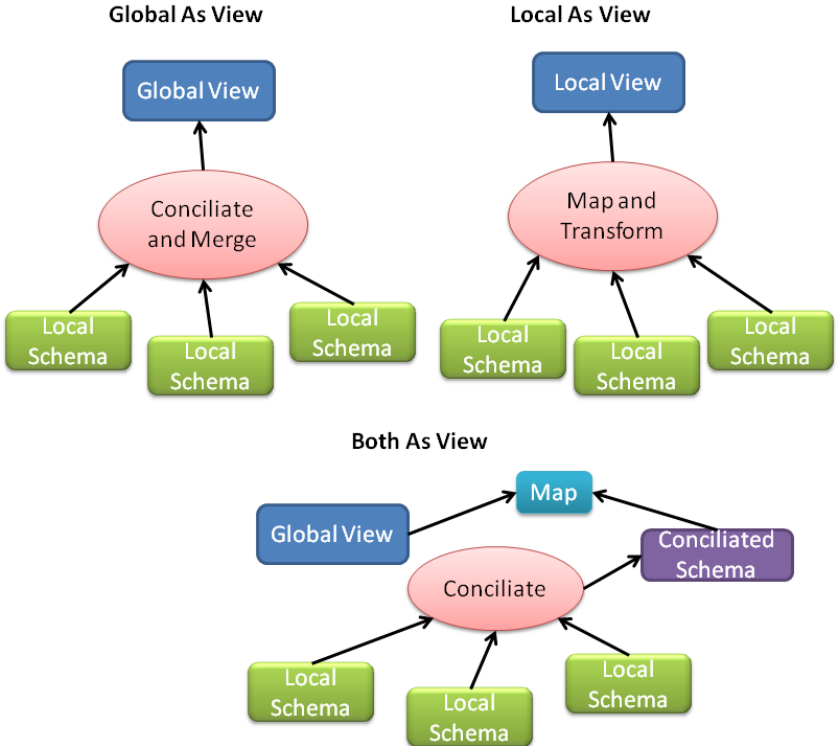


Figure 5 Global data view approaches [65]

2.1.2 Querying

Query refers to a transaction made in the database and can include the retrieval, updating and deletion of data. Here we are primarily concerned with the retrieval of information from public data sources. In the heterogeneous environment, the query method depends on the integration system. In general, a query is made to the integration system and the integration system processes it according to its implementation and returns the result. The main types of querying include

i) Direct query is used if the integration system has translated data from different sources.
ii) Query translation is the translation of queries formulated in a global data-manipulation language to equivalent queries in the specific language of the individual databases. Data are stored only in the constituent heterogeneous databases and when a query is issued against the virtual model, the engine for query translation and execution decomposes and translates the query into an equivalent set of local queries. Local results are then transmitted, transformed and combined to other data. All query translation techniques can be categorized in two main classes:
    1) Query translation based on procedural mapping where mappings between conceptual database schema and various underlying databases is performed through procedural functions that physically import objects from local databases into corresponding objects in the global environment. During this activity, mapped objects can be manipulated to improve query performances or access. Although simple in implementation, optimization cannot be done and all mappings have to be tested periodically and validated.
    2) Query translation based on declarative mapping specifies the correspondence between objects and operations at the level of global query model and objects and operations of the constituent query models. The representation of correspondence is encoded such that a software process may inspect it, and it is stored independently of the software code that performs query translations. A query optimizer can inspect and handle the declarative mappings by understanding the mappings' semantics of local databases.
iii) For integration systems where a global data model is implemented, the query is made according to the global schema and the system defines multiple queries for each data source. The results from individual sources are then compiled together and returned. Ontologies are considered synonymous of global conceptualizations and define the object classes, relationships, functions and constants for the application domain. Ontologies are similar to query models since both include a formal abstract model for representing properties of objects in a domain, a definition of the object classes and of the relations and functions of the members of those classes and a specification of the object constants.

2.1.3 Analysis support

As the size of the data grows separate analysis steps being performed in isolated manner becomes cumbersome, as large data needs to be moved to different locations especially in a distributed environment. Thus, the analysis tools should be plugged into the system so that the analysis may be performed in an automated setting with a synchronized meta-information being maintained for later use. Galaxy is an example where many analyses can be performed on the same platform but provided very limited management functionalities. As the analysis tools are dependent on the type of analysis and change as new improved tools are

made available, for the analysis the system should support a pluggable interface such that tools can be attached based on requirements. Further workflows and pipelines are a common feature in present computational experimentation. A workflow includes multiple steps being carried sequentially using different tools in order to perform a certain analysis. Integration of workflows as an analysis support is indispensible for any management system.

2.1.4 Visualization

Genome browsers are the main visualization tool for the graphical representation of the genomic data for browsing, searching, retrieval and analysis. Without a visual support understanding the data and drawing meaning is a daunting task for many. Within the browser, comparisons should be carried out between experimental and public data. As nowadays, genome browsers are primarily web based, providing support to analysis and management systems as well. The initial genome browsers, including the NCBI map viewer (vertical visualization of annotation), UCSC and the Ensemble genome browser (horizontal visualization of annotation) provided only a comprehensive set of tools for the data visualization and information retrieval. With the completion of the Human genome project and other Genome projects there has been a huge increase in the availability of genomic data. Thus the annotations present in the versions of genomic browsers have increased many-folds. Nielsen et. al [66] provides an excellent review on visualization of genomes and divides visualization into

i) Visualizing sequencing data- analyzing both in the context of de novo assembly and of re-sequencing experiments. The primary concerns in this category includes the alignment of large number reads errors in results of the assembly. Mainly, closing gaps, correcting mis-assemblies and improving the error probabilities of consensus bases can be corrected at the finishing step. Mostly automated, but sometimes manual editing are needed, thus is the requirement of the ability to edit contigs.
ii) Browsing genomes- both annotations and experimental data mapped to a reference genome. The primary purpose is to provide a graphical display of the annotations and the ability to compare experimental data with available knowledge. The different types of annotation data are stored in different types of tracks and often with different display modes relating to the type of information needed. Some browsers also provide database querying facilities and interfaces to link analysis tools. Next generation browsers mark a major shift from the centralized data model where all browser data preparation was carried at the server. In the de-centralized node, the data are maintained at the server but the visualization is rendered and viewed at the client side, thereby reducing the server load. JBrowse [67] is the first browser to implement a de-centralized model for visualization.
iii) Comparing sequences from different organisms or individuals. The growing number of complete genomes has created the opportunity to compare the genomes within specie or among multiple species, for identification of functional elements, studying rearrangements and evolution.

Visualization methods in these domains are at different stages of maturity, still integration among the various visualization tools and to the analysis software can provide better use of genomic data for better research in biology.

2.1.5 Management

The primary concern for high throughput research laboratories are the management of the voluminous data being generated. Intelligent data management involves the understanding of the data type, the generation technology, the metadata descriptions concerning the data, and relevant analysis tools or workflows concerned with each type of data. Further, the access to the data needs to be controlled. Collaboratory research emphasizes the availability of data at different location, thus within the system the transfer of large data files needs to be efficiently handled. All aspects related to data are mediated by the management module with access control and requires a project management. Incorporating aspects related to projects to which the data belongs, first segregated the data according to the projects, secondly it creates a hierarchical division of responsibilities and data usage.

Project management consist of supporting the project from the starting to its successful completion and using the management data collected during the course of the project to guide the people for a successful outcome. Managing a project effectively means thinking before acting, identifying and dealing with potential problems before they occur, and constantly monitoring to determine whether your actions are achieving their desired results. The primary components include the planning, organizing and controlling parts. The planning section involves the identification of desired outcomes, the collaborators interested in the outcome, the activities involved for completions, expected tenure of each part and the risks involved. Based on these observations the people involved are assigned roles and tasks. Finally control is needed during the life of the project. Control is essential for the assessment of the project performance at each step with the identification of positive contributions, managing failures and risks and keeping everyone informed of the projects status.

2.1.6 Reporting

With high degree of automation of experiments and analysis, there is a need for each action to be recorded and reported. The results of analysis, errors and system information need to be corresponded as well. Thus there is a greater need for the reporting and tracking module to function closely with the management module. The reports also serve in representing the advances of the research and findings in a coordinated manner. Projects require a complete documentation of each phase with a complete report of deliverables and results. Reporting module also help in the maintenance of transparency among the collaborators and through a dynamic reporting module the results of analysis and important findings can be easily circulated and made available among all partners. Dynamic reporting module also

simplifies the process of project dissemination as all information is already organized in and available to support the report creation.

2.2 Dynamic pipeline creation and management modules

Pipelines are an indispensable approach in today's complex research. As the tools are becoming specialized in performing a particular task with high accuracy, for flexibility of analysis and fast evolution of the biological fields workflows provide the flexibility to researchers. Present high throughput data analysis requires high computational power and storage. Small to medium scale research laboratories rely on distributed architecture for making analysis feasible.
Saha et. al. [68] identified three important capabilities of such an orchestrating system:

i) Flexible architecture so that one software system can be used to analyze different data.
ii) Allow the inclusion of new tools in a modular fashion so the software architecture does not have to change with the addition of new tools.
iii) Facilitate data integration of analysis results from different tools that were computed on the same input.

For a dynamic system to function in a distributed environment the following modules can be identified:

2.2.1 Pipeline creator

The primary function is to create pipelines quickly, efficiently and having the flexibility to add analysis tools as required. The created pipelines need to be maintained for reproducibility and availability for multiple experiments. Linking various analysis tools with the system can be challenging as the tools may differ in their installation and working. Thus, metadata related to each of the tools needs to be correctly maintained in order to provide detailed descriptions about the tool and its usage. In order to avoid unrelated tools being linked together in a pipeline creation process, suitable checks and rules are needed. The implementation of the creation process can be through the use of wizards or graphically with the pipeline information being stored in some repository. The repository may be a dedicated database or descriptive XML files.
Implementation of the pipeline logic can be broken down into the following generic segments:

i) Loops represent the repetitive part. Repetitive tasks include the batch running of the pipeline for a number of samples, repeating an analysis or processing step under a given set of conditions, etc.
ii) Conditions indicating the flexibility of the pipeline with respect to the different types of results. Based on an intermediate output the path of the pipeline may be altered and such decisions are implemented through this construct. Such a construct is especially useful when errors occur, describing an alternative process for analysis or excluding that particular data from the analysis.
iii) Execution node related to the running of an analysis step.

2.2.2 Data management

In a distributed environment, data needs to be made available for the workflow to proceed and upon completion the results are transferred for storage. Thus for running multiple workflows, data management becomes relevant. Data are transferred from the storage server to the analysis server for processing based on the requirements of particular workflows. The quality of data received and the validation of results before being archived is essential for automation systems to avoid errors. Metadata information about the data helps in the evaluation of the availability of resources needed for the analysis to proceed.

2.2.3 Pipeline management

Pipeline management incorporates the running of the pipeline, its reusability and editing. The relevance of the management module is well explained throughout the subsequent scenarios:

i) Each analysis server may contain varying number of available applications and analysis tools, thus supporting a limited number of pipelines. For each pipeline execution the correct server needs to be selected. For identifying a server for execution of the pipeline, a greedy approach is helpful. This means that the first available server that supports the pipeline is selected. This approach uniformly distributed the work load over all the available resources.

ii) Manual control over the pipelines is enforced through this module. The controls include pause, restart and stop. These control commands are coordinated to the relevant changes in the running pipeline. Information related to the current stage of the executing pipeline is used for such control options.

iii) If the pipeline is to be upgraded or its parameters changed, then there is a possibility of errors as old runs may have inconstant results with the present change. Thus a control and proper description of changes should be present within the system for users to refer.

The pipeline management module manages the availability of resources and controls the execution of assigned tasks. This module should also support the scheduling functionality for jobs predefined for the future. For the execution of workflows, following processing model types can be implemented:

i) Data flow driven – Data availability is the prime concern. The workflow waits until the data is made available and only then the current step is executed.

ii) Control flow driven – The steps are executed upon receiving the control signal and the time of arrival is the important factor.

In scientific workflows, data is valued more and hence the data flow model intrinsically is implemented with an event driven control of execution. Thus the

execution of an analysis step needs to be started only when the data is completely made available.
Role based access control for the system enables control over the features provided to the different users. The access control also provides a means of screening out malicious access to data and servers. Further data safety can be maintained through an independent internal server with access to only the system. A report generation system should be integrated with the management module for providing a summary of the actions and results of the analyses.

3 Technological requirements

The surge of data volume in the post genomic era has called for unprecedented measures to be enforced in all aspects of genomic research. The impetus for improvements is felt most in the demands for technological upgradation. The effects “big data” can be felt in the technological arena as well. This chapter covers the available technologies that have the potential to provide solutions to the current needs.

3.1 Programming language

The process of creating any computational tool requires the identification of the programming language. Choice of a programming language differs according to the purpose and implementation. The initial tools in genomics were primarily web based. Perl dominated the server environment of these tools with the specific tools written in C/C++ language. As the size of data and the complexity of analysis increases, there is a conscious effort towards the use of both the web and the desktop environment. This means users of the newer tools can access through a web browser and installed programs. The use of programs installed on user’s computer increases the user of user resources and thus improves performance. Thus present programs need to be independent of the user computer’s operating system. For this reason Java is becoming a language of choice. Further Java inherently provides security through the object oriented approach. In Java programs are written in modules thus increases code reusability. Open source community provides extensive support to the development of packages in Java, thus increasing the diversity in usage of the language. For web programming, a number of java based tools are available that can interact with java classes to provide dynamic and versatile features. A notable platform is Ajax based Zk [77] tool.

3.2 Software architecture

Software architecture represents a general description of parts of the system and the modes of communication between them. It broadly describes the strategic layout of the global requirements of a software system. The requirements of genomic research are extensive in terms of technology and thus a management system requires an elaborate design to support research requirements. Thus out of the various types of available architectural framework, in a robust representation, the Enterprise architecture, is expected to provide the required design support for the complex genomic research environment. For genomic research environment, Enterprise architecture can be defined as the organizing logic for processes and infrastructure reflecting the integration and standardization of the research center’s operating model. The operating model is the desired state of research process integration and standardization for the delivery of results and other related services. In general the Enterprise architecture can be divided into the following layers:

i) Processes and activities
ii) Data collection, processing, storage and distribution
iii) Application layer consisting of tools used within the system
iv) Technological layer representing the devices and environments providing users' access.

For the implementation of an architecture several design patterns are available that solves and delineate some essential interconnected elements of a software architecture. These architectural patterns can divided into the following sub domains

i) Data Architecture- refers to patterns used for defining a particular target state and the subsequent planning needed to achieve the target state.
ii) Data Integration or service oriented architecture- represent patterns that target the unification of the data or services into a common format and layout. Sometimes also referred to as Enterprise information integration.
iii) Master data management- covers all aspects of data maintaining consistency and control in the ongoing modifications and application use.
iv) Data modeling- involves the data representation at different architectural levels. It represents the developmental pattern from a conceptual model to logical model and the implementation model.
v) Business Intelligence- in its broadest definition incorporates all methodologies used in the conversion of unprocessed data to gain better insight and provide support to the decision making process. In this form it incorporates all the above domains as well. However business intelligence is primarily involved with the presentation of information and decision making support.

Sub-Domain	Architecture Pattern	Design	Solution
Data Integration/ Service Oriented Architecture (SOA)	Data Extraction Transformation & Loading (ETL)	• Change Data Capture • Near Real-Time ETL • Batch ETL • Data Discovery	• Error handling • Job scheduling • Data validation • Slowly Changing Dimensions Load
	Managed File Transfer (MFT)		
	Enterprise Application Integration (EAI)	• Publish/subscribe • Request/reply • Message Exchange Patterns	• One-Way • Synchronous Request/Response • Basic Callback • Claim Check

Data Architecture	• Transaction Data Stores (TDS) • Master Data Store • Operational Data Store • Data Mart • Data Warehouse	• Custom Applications Databases • Packaged Application Databases	
Business Intelligence	• Transactional Reporting • Operational Reporting • Analytical Reporting	• Transactional Reporting Data Access • Operational Reporting Data Access • Analytical Reporting Data Access • Analytical Dashboard Data Access • Operational Dashboard Data Access • Data Mining	• Real-Time Dashboards • In-Memory Analytics • Statistical Analysis • Predictive Analytics
Master data management	Master Data Hub	• Master Data Replication • Master Data Services • Master Data Synchronization	
Data Modeling	• Dimensional Data Modeling • E-R Data Modeling	• Modeling Standards • Naming Conventions	

Table 1 Tabular description of characteristics of the sub domains in architectural pattern design (adapted from [69]).

The demarcations of boundaries among these sub domains are fuzzy and thus patterns can belong to multiple sub-domains depending on their implementation. During implementation of a software system multiple patterns get involved in individual sub-systems. Still there are other patterns that cannot be linked to any particular sub-domain. These patterns are conceptual in their implementation. These patterns include implicit invocation, model view controller, multitier architecture, Peer-to-peer.

Observing the sub-domains in the context of a management system for research, we find a representational similarity of the parts of the overall problems. Individual data sources and technologies needs to be integrated with various data sources. Overall data management with models of increasing information content follows

next. Finally the development of knowledge is achieved through analysis of divergent information and decision making.

3.3 Database

A database is a collection of data. As the data grows, for its management specialized software, database management system (DBMS), is integrated with the database. Execution of all commands to the database is controlled by this software. A database system can be commonly divided in flat file database, relational database containing primary and foreign keys to maintain a relational model and object oriented relational database. While generally the flat file database is slow, the relational systems works with the use of the structured query language (SQL) and data transactional language (DTL). Every database transaction obeys the following rules (ACID):

i) Atomicity- The transaction is either complete or incomplete. This rule implies that incomplete transactions do not make partial change.
ii) Consistency- after any transaction the database should be in a stable state.
iii) Isolation- Every transaction is independent and the changes made to the database through a transaction is not noticeable through another transaction.
iv) Durability- Database transactions should be safe in all situations including a system crash with all modifications are stored.

With the object oriented relational model high level languages can access data efficiently and hence provide greater flexibility to the representation and utility. The most popular relational databases are MySQL, PostgreSQL, Oracle, DB2, SQLite of which postgreSQL is an open source initiative. The database consists of multiple tables containing data records at each row and fields in columns. A database schema is the relational description of the database and is maintained through the meta data related to the tables.

Due to the recent surge in the volume of data a number of No-SQL data bases are being developed. These include the MongoDB, memcached, Redis, CouchDB, Apache Cassandra and HBase. Another subset of No-SQL databases are the xml databases that include BaseX, eXist, MarkLogic Server, MonetDB/XQuery, Sedna.

3.4 Network connectivity

In a distributed system, where the physical location of the computer is diverse, communication and connectivity becomes the core for the system to function properly. Security, speed of transfer and integrity of the data being transferred vary among different protocols. A network protocol is a system of digital message formats and rules for exchanging those messages between computer systems. The popular protocols include

i) Transmission control protocol (TCP)
ii) Hypertext Transfer Protocol (HTTP)
iii) File Transfer Protocol (FTP)

iv) Post Office Protocol (POP3)
v) Simple Object Access Protocol (SOAP)
vi) Secure Shell File Transfer Protocol (SSH or SFTP)

SSH provides a secure channel for encrypted transfer but is slower than the most common protocol for file transfer FTP. Requirements of the transfer of large data over the network have led to a numbers of new protocols being tried; however they are at an early stage of development.

4 System development and implementation

This section represents the description of the development cycle involved in the creation of the software system to attain its working goals. Based on the general requirements described earlier, this section gives a finer view of the various parts that coordinate to provide the working environment and functionality.

4.1 AD2BioDB

AD2BioDB, as a clinical genomics data management system can be broken down into many layers of development. These developmental layers are the logical separation of the system into sub-systems. A summary of the objectives of such a system should help to better understand these layers. These objectives include:

i) Develop a flexible system that supports the present experimental technology as well as that of the future, and has the ability to connect other tools for providing uninterrupted research environment.
ii) Provide user access through the web as well as via the stand-alone desktop client program.
iii) Manage and store research data in an automated setting.
iv) Provide security to data by controlled user access, controlled data transactions and logging of each transaction.
v) Reporting of results, status of system processes and alerts for errors are supported features for a complete implementation of working system.
vi) Concurrent support to multiple databases is essential for research due to the distributed nature of public data sources.
vii) Supervised data import export.
viii) Dynamic searching for relevant data.
ix) Supporting an environment for implementation of available analysis.
x) Workflow creation and running requires an inbuilt distributed workflow management support.
xi) Graphical support for better understanding of results and visualization of experimental and public data.

These objectives can be divided in structural layers:

i) Application platform represents the core functionalities of the software framework. It is static in deployment and is governed by system specific immutable rules on which all dynamic functionalities rest.
ii) Client management is primarily involved with the modes of access by the users, that is, web or programmatic access. This module also supports the concurrency among multiple accesses.
iii) Data Integration deals with the availability of data from diverse sources as a unified stream. Data available through diverse repositories are combined to a common format and presented.
iv) Functional layer is the most dynamic segment of the entire system. Here features can be added, modified and revoked. Temporary operational and presentation rules can be defined at this layer.

4.1.1 Application platform

Genomic data management system requires a distributed client-server enterprise environment for its implementation such that storage and analysis infrastructure can be accessed from a central system server and the user's access is limited to the requests made to this application server. The client systems primarily format and display information for the user. In this setting the infrastructure is secured and any change made will be invisible to the users. Enterprise architecture is essential due to the complexities of the overall processes involved. The need for a dynamic system, with heterogeneous integrated data and subsequent analysis, has multilevel complexity as well as the management module justifies the efforts needed to implement enterprise architecture.

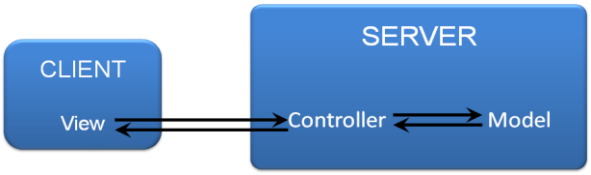


Figure 6 Schematic representation of Client-Server architecture with MVC design

The general architectural pattern for implementation requires security. Therefore the model-view-controller design paradigm [70] is a preferred choice as the user access is further limited to the view part, meaning that the application logic is maintained only within the server and the displayed interfaces are only exposed at the client side. User requests are made through the interactive views. Each request is received and processed by the controller, which performs the management of requests and calls appropriate model to execute a response for the request. The model part represents the location of the functions that implement the actual programmatic logic and infers functionality. This software design method minimizes the availability of portions of the software system for external connections. This method of designing has an added utility; as the view part is separate from the programmatic logic of the model, reusability of graphical components is possible. This is advantageous as the presentation time significantly reduces.

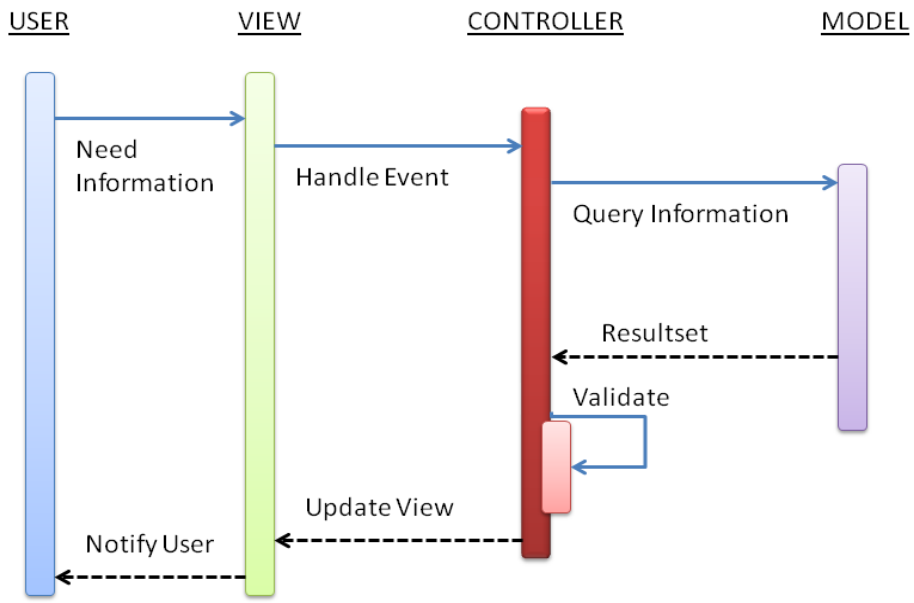


Figure 7 Detailed representation of MVC working

The server side application is made available to clients through Apache JBoss server [71]. The JBoss server provides java enterprise standard. It acts as a container for the functional java ([72], [73]) classes and provides connection ports industry standard. It provides many preconfigured features like enterprise java beans [74] and java messaging service etc. which can be used as required.

Architectural flexibility means that functionalities can be added when required with minimum modification to the platform. This is achieved by storing the application data in a database with a unified structure. Adding functionality is carried out by updating the database with relevant information. The application architecture follows a common logic in providing all functionality. This abstraction is the basis of functional flexibility. A database driven application also confers robustness in management and performance. This application type requires a tight coupling of the database with the application model such that the model can accurately formulate and execute database operations. Therefore the model can be regarded as the overall collection of database schema wrapper classes that links the relevant application processes. Thus database driven applications provides persistence and stability through the wrapper mediated data control.

Client server communications is provided through the use of enterprise java beans. Beans are of two types, server beans and client beans. Processes that are to be run on the server represent the server bean type, such that whenever such a process is to be performed, the instructions are synchronized with the server component of the process and the execution takes place at the server. On the

other hand simple beans work and are present in the cline side. Java messages can be used for interactions related to the unstructured data. Unstructured data or unformatted data related to those data that are related to the system directly, are present as supplementary information. This can include communications among the users regarding a particular analysis or reviews of the supervisor.

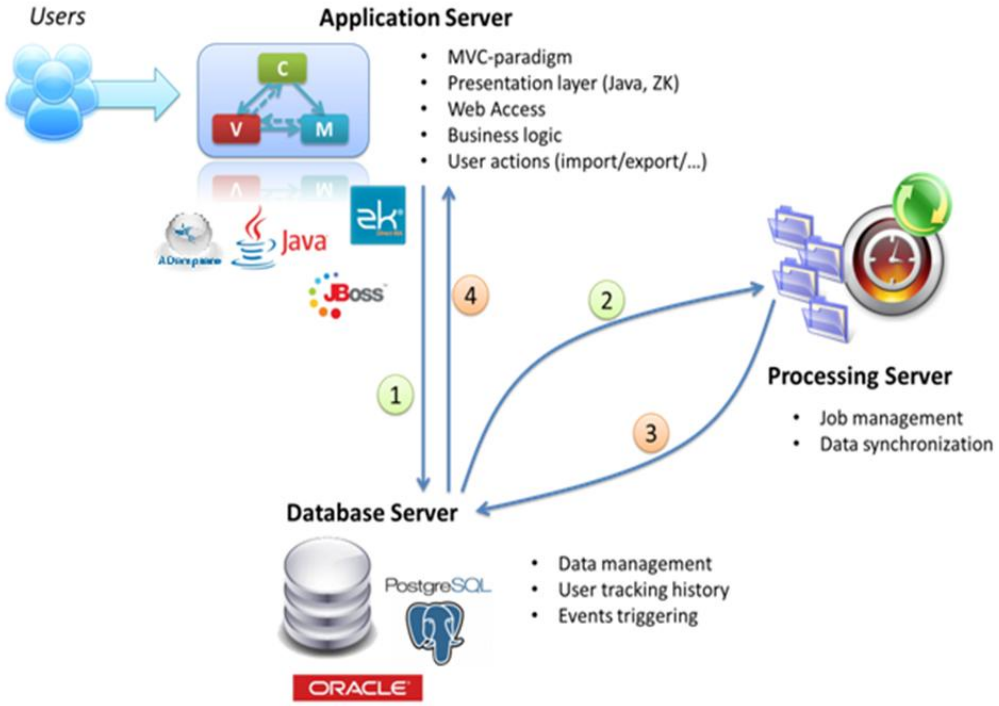


Figure 8 AD2BioDB architecture

ADempiere application framework

The ADempiere ERP [75] environment is a unified collection of software components. These components co-operate together to produce a unified system. Enterprise Resource Planning (ERP) is an industry term for a range of services that provide functionalities to a business. ERP systems integrate all data and process of an organization into a single system. This is obtained from a unified database to store the data from various sub system modules. These functionalities help to manage the important parts of a business, including product planning, maintaining inventories, interacting with suppliers, providing customer service and tracking orders. ADempiere software acts as a horizontal platform where other or new vertical applications can reuse or sit on it. A successfully introduced or migrated vertical application can utilize the integrated multiple rich interfaces and application engine. Intending application developers have to realign their Software Business Model to an entirely new one. ADempiere is open source software under the GPL

license [76] meaning that all portions of the application can also be modified and reused according to the requirements of the new application.

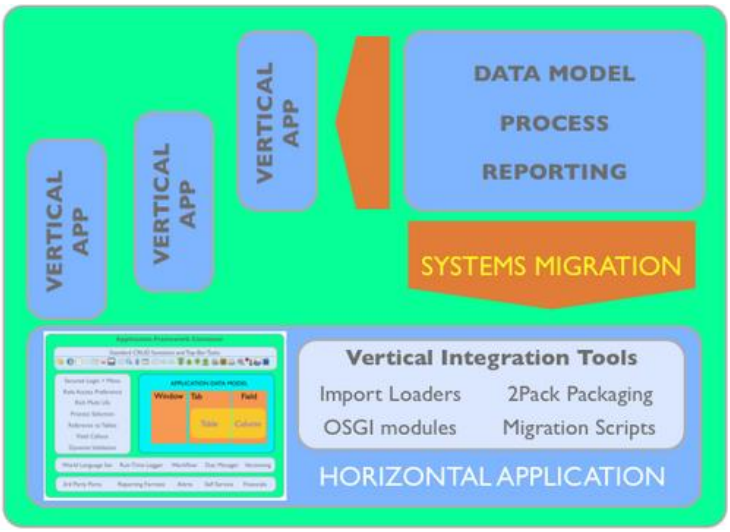


Figure 9 ADempiere- platform for rapid software development [75]

As application platform, for a rapid application development ADempiere provides all the technological requirements for the development of AD2BioDB. The support provided by the application framework includes:

- The use of this platform also provides logging and tracking capabilities, essential features for a good management system.
- The use of application dictionary provides dynamic capabilities for a coding free creation and modification of the presentation components viz. windows, trees etc.
- Reporting structure is inherent in the framework and can be extended for the implementations specific to the requirements of AD2BioDb.
- Database connectivity in a secured environment is another feature, relevant to the needs of the biological data management. Database management scripts are defined in the framework for user in all implementations.
- The support of the framework is available for most operating systems as well as for web browsing.
- Error tracking is an additional feature available for developers and system administrators.
- The platform also provides basic support for workflows and processes to be run at the application server. This feature can be extended for the present distributed analysis environment.

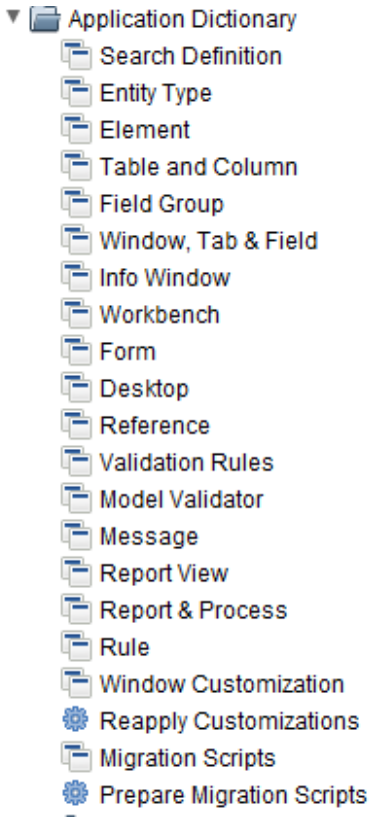


Figure 10 Application dictionary for the dynamic configuration and maintenance of software parts

The key feature behind all the flexibility provided by the application framework revolves around the application dictionary. The application dictionary is meta-data driven implying the contextual data shapes the outcome. The tables and columns item within the dictionary is the fundamental component that connects the underlying database to the application framework. The database can be modified through this dictionary item and are synchronized with the application. Elements represent the terminologies used for the data elements. Element is also related to the translation feature. Validation rules, defined in the context of a column field, are dynamically verified based on the predefined rules or user context, at time of rendering the data. Reference refers to database column field types that are either Data Types (i.e. Integer, Date, Time, image, hyperlink, etc.) or a List validation (i.e. user pre-defined dropdown lists) or Table validation (i.e. drop-downs for table key columns). All windows are defined in a standard dynamic way by reference to the defined application dictionary. This application dictionary window thus relates to setting up the Windows, and the Tabs (sub-linked windows) and Fields that are displayed on those Windows relating to the metadata maintained within the database. Window can be of window, form, report and process, workflow types. Info windows are used for quick searches and information views. Where database views may exist within the underlying database, the report views represent the

Database views in the system in order to be accessible. Reports and processes are used to set up reports (link to a Report view) or a process that can link to a Java code class. Reports and processes may have parameters that define a selection process. If a report is also displayed as a Dashboard then an underlying dashboard widget needs to be defined.

4.1.2 Client management

Client for the application refers to the modes of establishing access and includes web and programmatic access. Desktop client programs can connect to the application server, through the client access ports. Client is an interface to the other server layers. The client only allows communication to take place between the end user and the application server. The client presents the view to a user and tracks the conversational state between the server and itself. The Client Tier is restricted by the network so client should only connect to the server when it has to, transmit only as much data as it needs to, and works reasonably well when it cannot reach the server. J2EE [73] platforms generally encourage thin-client architectures. Using a browser client however provides low response times. The client depends more on the application server for presentation logic, so it must connect to the server whenever its interface changes. This causes the browser client to make many connections to the server which will be a problem at slow connection speeds. Java based web programming language ZK [77] is used for the creation of dynamic web part. ZK is the leading enterprise Ajax framework [78]. ZK offers a rich internet interface [79] and fast development environment enabling UI designers and business analysts to come together and make on-the-fly changes to the UI. ZK supports all major patterns of development such as MVC, data-binding, templating, etc. Through ZK, all functionalities of the application can be easily presented for availability through the web interface.

4.1.3 Data Integration

Data integration in genomics involves unification of public annotations available through diverse data sources and utilization of this information for the description of experimental data. Genomic annotation data repositories mainly include UCSC [80], NCBI [81]], EnsEMBL [82] and dbSNP [83]. The UCSC grants access to the genomic database via FTP (to files and tables) and directly by MySql ODBC. NCBI created access to data via API, the eUtils, with more programming languages such as Perl [84] and Java. Moreover EnsEMBL exposes a good system of web-services, very useful also for workflows. Data about SNP can be downloaded from dbSNP, which allows integrating data about chromosomal and contig position, heterozygosity, alleles and function of the related DNA portion. Other public collections don't have programming interfaces or accession systems and thus they need custom wrappers, provided by third party community tools and libraries already created for this purpose or implementing new ones. The final data fusion process provides a way to recognize and make explicit conflicts among the same biological objects and it presents a complete and consistent representation as a whole. While schematic conflicts (dealing with different attribute names or

differently structured data sources) and identity conflicts (related to different way of identifying a real world object) are considered in the integration process, the data fusion activity focuses on explaining multiple representation of the same biological object.

4.1.4 Functional Layer

All the previous layers provide support for handling the dynamic components and thus are generic in implementation. The functional layer consists of the actual representation of the various working features of the software and interacts with the data stores through the platform. This layer provides the dynamic addition of feature utilizing the support of the platform. The implementation of functional layer can be better discussed in a modular representation [Figure 11]. The ADempiere framework provides the generic management support for working of the entire application. The functional modules enhance and implement the specific features required by users in different project implementations.

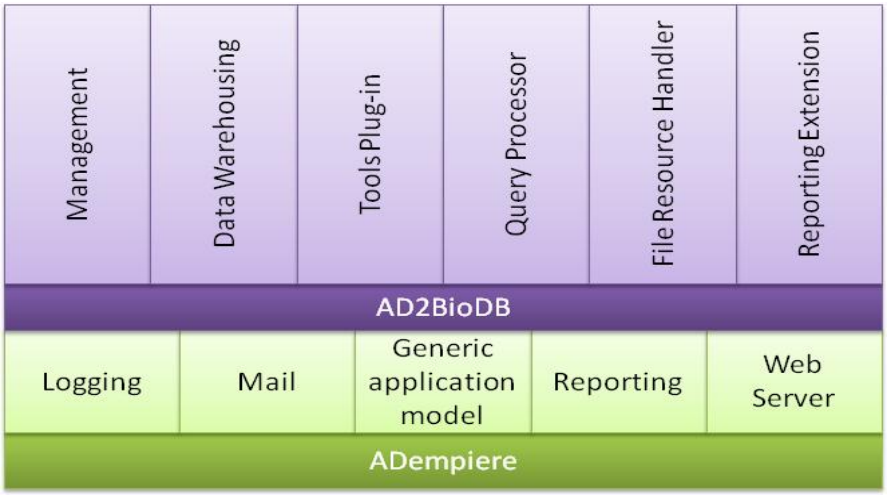


Figure 11 Modular representation of AD2BioDB

Management

Implementation of management is primarily based on access control. Through roles controlled access is imposed and includes system administrator, supervisor, data handler, data analyst and collaborator. System administrator supervises the working of the entire software and makes available new features. Supervisor role relates to administrative functions within a particular project and has access to all features. Supervisor receives all reports related to the project and can add data handler, data analyst and collaborators. Data handler role is data import into the system. Data analyst analyses the data by the use of various workflows made available for the project. Changes or upgrades for the workflow are enforced by the system administrator. Report of the results is made available to the data analyst. Collaborator represents any person who does not participate directly in the project

and is provided access to the summary of reports. Figure 12 represents the access level provided to each role. Any user can be associated to multiple roles for different projects. Selecting the project provides access to the related laboratories. Thus at the time of logging into the system, purpose and resource requirements of the user are identified and hence enforcing security.

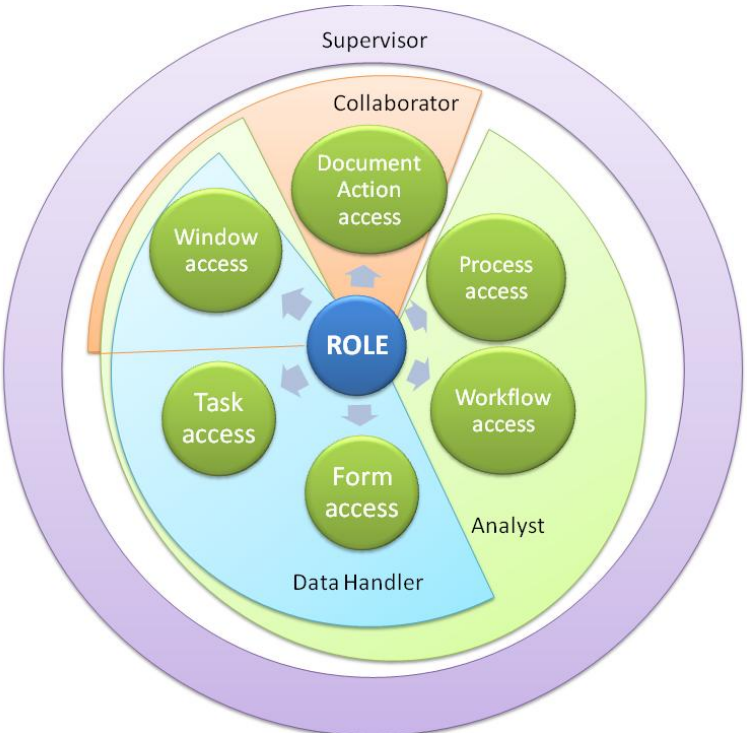


Figure 12 Controlled access through definition of role

Project is the highest level of independent functional entity and has one or more research centers involved [Figure 13]. Project contains the organization tree, where the complete set of tasks is divided into phases and sub-phases. Tasks can be analysis, data generation, deliverable, etc. A task of type analysis includes actual experiments conducted. These experiments can be iv-vitro or in-silico. The term in-vitro experiment is used in a broad sense encompassing all experiments that are not performed in-silico. Computational experiments consist of processes and workflows, related to the technological platform used in generation of data. Each project defines the technologies associated and definition of reporting structure. The selection of a technology at the time of the creation of the project phase includes all resources related to that technology, for example, the data formats involved, availability of dedicated servers for the execution of analyses and workflows, storage etc. Dynamic reporting is done for progress, resources used and deliverables. All nodes in the organization tree have the progress&status tracking, issue tracking, deliverable features. The status is calculated based on the proposed and actual dates associated with each part of the project organization. At

the project and phase levels the status of the next sublevel is considered for the status estimation. Issue tracking is associated only at the level of the tasks.
Further, the project may contain phenotype tree containing clinical information and can be used for the screening of data for specific experimental studies. Collaborations with partners can be maintained at the project as well as phase level. Subjects under study in a particular project contain demographic information and are separated from their actual data, hence ensuring confidentiality. Access to demographic information is restricted while the data is available for analysis. In certain studies example Cohort studies, family studies, subjects are segregated into multiple groups depending upon certain characteristics. Such groups need to be identified with the project definition. The creation of the project is enforced by the system administrator. The project requirements has to be predefined for project creation in a comma separated file (.csv extension) while project specific features can be added later by System administrator. Users related to multiple projects and with supervisor role needs to be added by the system administrator while project specific user for data handler, data analyst or collaborator role can be added by the project supervisor.

The system can be configured to manage multiple technologies needed in particular projects. Technological definition is aimed at the management of data imported into the system. Within technology the type of data, formats used, the handling and processing information needs to be defined. This procedure helps the system to understand the differences in the heterogeneous data even when present in the same formats. The change in formats for any technology can be adapted by updating the details for the specific technology.

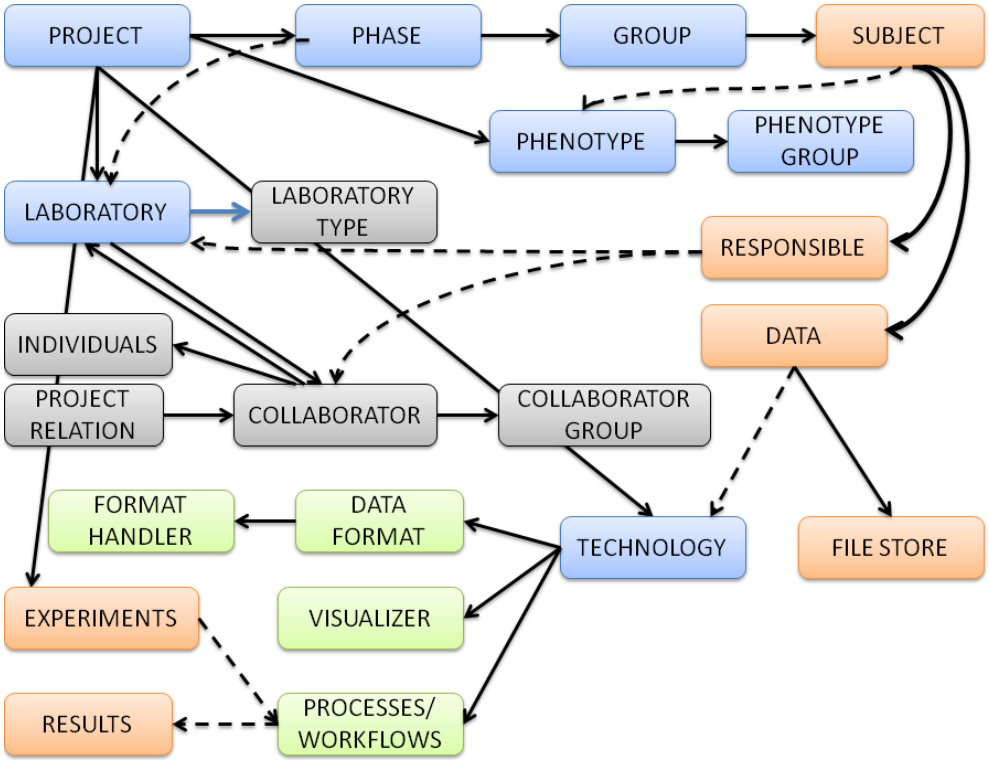


Figure 13 Functional design of AD2BioDB: Boxes in blue are defined at the time of project creation. Grey boxes indicate supplementary information related to project management. The orange boxes represent demographics and data. Green boxes are defined at the time of addition of technology within the system but can be modified as required.

Data Integration

Data from multiple sources are made available for integration through dynamic data source addition functionality. As described earlier, data repositories can have diverse mode of providing data access and hence for a dynamic functionality, each access method needs to be included separately and in a generic form. Presently the heterogeneous database access through SQL queries has been implemented as Global data model with local as view approach. Mediator based architecture [Figure 14] implements the mapping process. While the individual databases are accessed through wrapper classes, at the integration layer integrator class is responsible for mapping between the generic global schema and the database specific local schema. Mediator is also responsible for providing the unified annotation information in a global presentation.

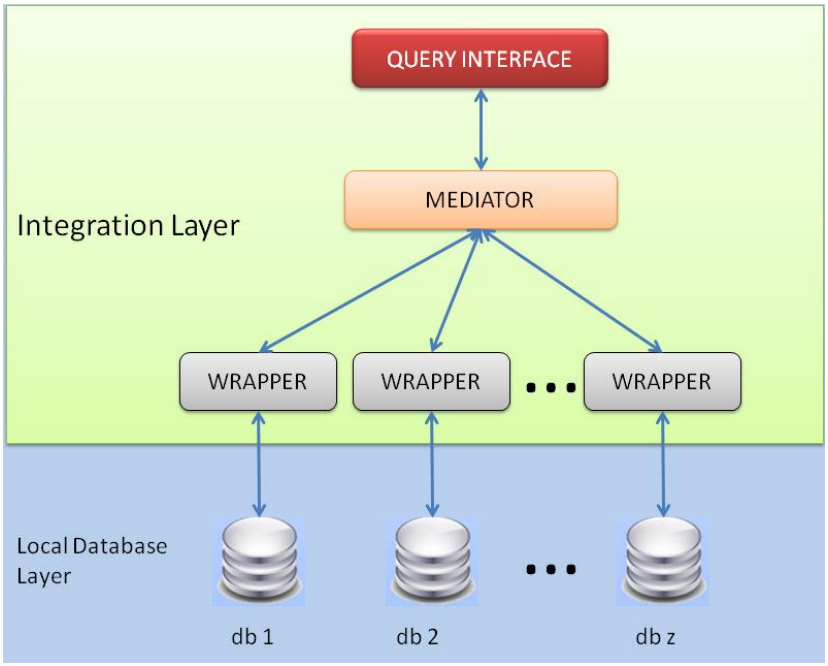


Figure 14 Description of Mediator based architecture for data integration

The experimental metadata is contained within the application database and consists of following sections:

i) Project section consists of information related to the project management structure. A project may contain phases and sub-phases with data groups. A project is associated with a primary laboratory but can involve multiple collaborator laboratories with a defined relation within the project. The collaborators can be assigned varying levels of access and priorities.

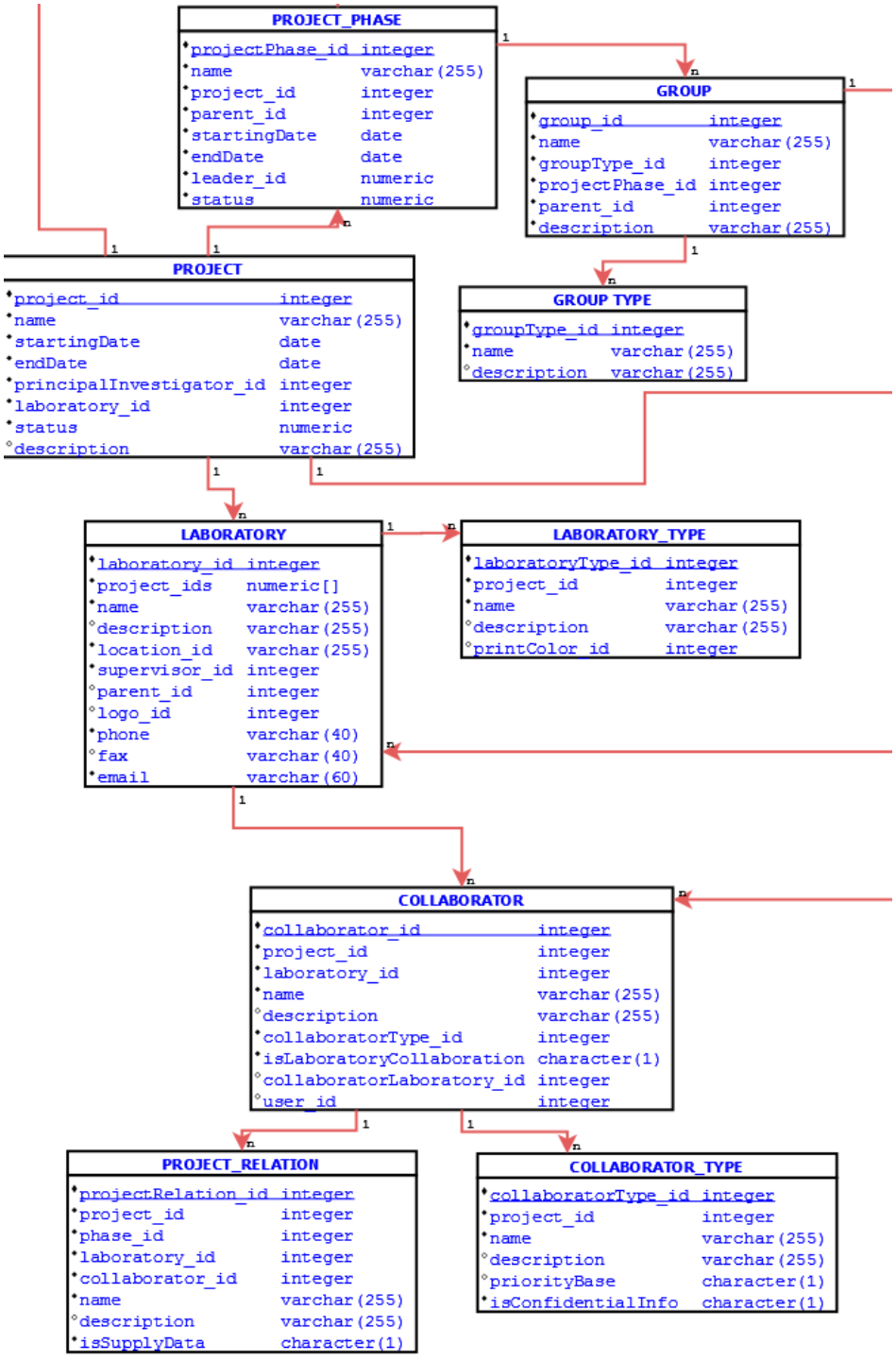


Figure 15 Sample database schema for Project structure

ii) Subjects contain demographics and data related information. The actual data is made available through this section. The project phase is associated with the data through the groups. The general information of a subject is separated from the confidential demographic information. The data table contains information about the data and the data generation information. All subjects have a responsible person who has access to the demographics information table. The responsible is part of the primary laboratory or is a collaborator. The schema is general and applicable to all species, humans and others.

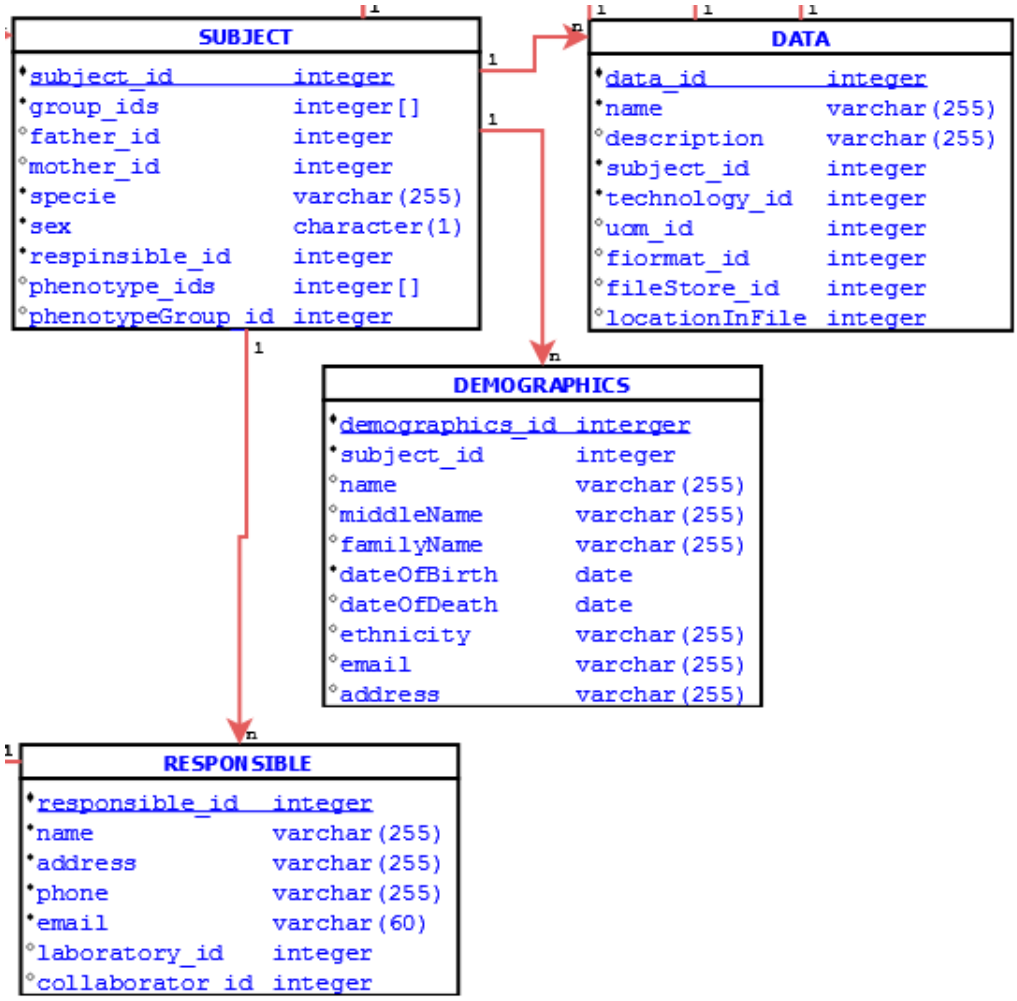


Figure 16 Sample database schema for demographic structure

iii) Technological part relates to the supplementary information for data usage. The common information related to a particular technology is present here while technology specific information can be placed separately linked by the

'extendTableName' column. All data formats are stored with information of how the format is processed.

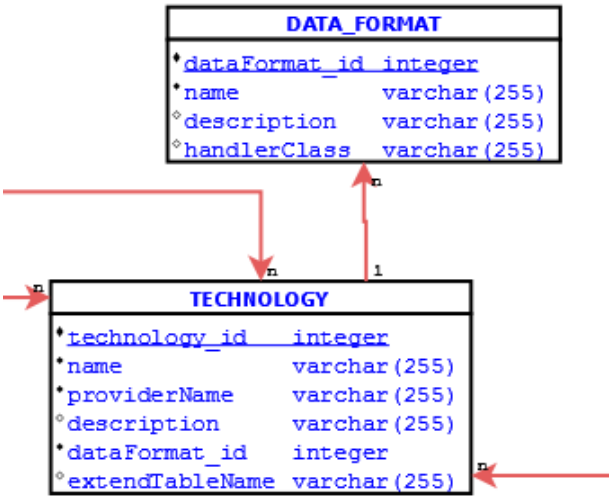


Figure 17 Sample database schema for technological structure

iv) Analysis relates to the experiments carried and the results obtained. The experiments are directly related to the project while the generic pipeline is associated with particular technology.

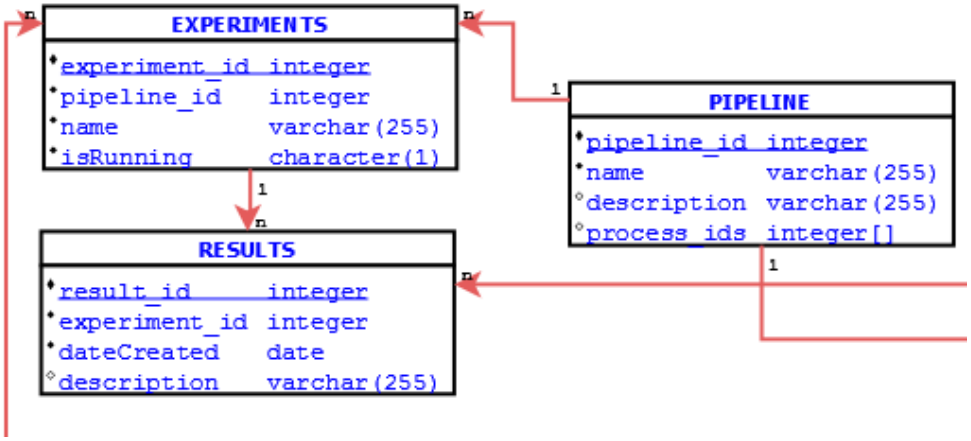


Figure 18 Sample database schema for analysis structure

v) Phenotype related to the hierarchical representation of the clinical observations and their units of measures. Phenotypes can be specific to projects and can be grouped to form complex clinical representations. Each phenotype is associated with unit of measure and a value. If the phenotype can be represented in two different units then the conversion formula needs to be specified describing the relation.

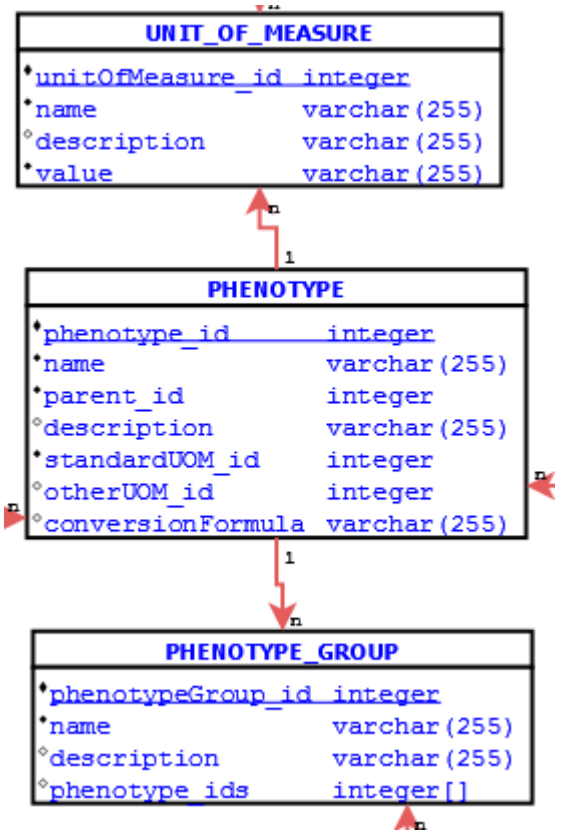


Figure 19 Sample database schema for phenotype structure

vi) Data can be stored in a secured dedicated server and its access is controlled. Such information about data locations are defined in the file store. Further the results of analysis are commonly file based and are also made available through this segment. Information about all files can only be acquired through this section.

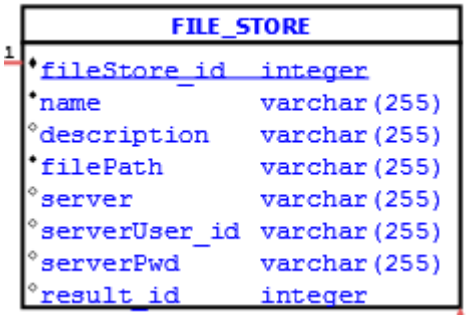


Figure 20 Sample database schema for phenotype structure

Query processor

The query processor, interacting with mediator class, takes user queries and carries pre-formatted structured queries to the individual data sources. The results are combined and presented for the user as a single table. The user interacts with the global schema and the application process all heterogeneity in the background. This functionality is at a basic level of development due to its dependence on the addition of data sources.

File resource handler

Genomic data is mainly file based and hence all experimental data are present in output files generated from laboratory instruments and analyses steps. These data files can be of different formats and thus their read write operations being carried out by different programs. For the proper identification of different format handlers in a unified manner, these handlers are wrapped into classes that extend a common abstract handler. Every wrapper class can impose the desired operation carried out by the specific format related tool. This abstraction method requires that the tools can be accessed either by the command prompt with the use of specific commands or has a programmatic access interface. The file resource handlers are associated with particular technology and gain access to the data files present in the data server through a secure channel.

Analysis

Analysis can be carried out by many tools and can also be complex involving the use of workflows, a collection of tools carrying analysis sequentially. Even for the same analysis step multiple tools can be used depending upon the users' choice. Thus the dynamic nature of analyses forces an abstract implementation of analysis. The incorporation of dependent tools and corresponding format handler are made at runtime. The tools can be added into the system by defining the location and the relevant command to carry out analysis. If the tool provides a programmatic access then a wrapper class needs to be defined for access. Although lengthy, it provides maximum flexibility to the use of the tool and should be preferred for tools that are frequently used. For the management of workflows, carrying multiple analyses as a single process requires many analysis servers. This complex process and its management have been developed as a separate system SeqPipe and have been closely integrated within this system. The implementation of SeqPipe is described later in this chapter.

Visualization

Visualization is the graphical representation of data such that greater sense can be mined quickly. Due to the diversity of the data within the system various visualization modes are made available. Technology defines the data and hence also needs to define the visualizer. In general the results are represented in a pre-defined format and are consistent over most projects. Therefore the default minimal

representation of visualization available within the system is the graphical elements within the reporting system. Further the public annotations are available through a common format and hence a genome browser for the graphical representation of annotations. Genome browsers are evolving following the technological changes and hence a pluggable browser is preferred to a static and integrated browser. This mode helps the system to adapt to future technological improvements.

Reporting

The system platform provides the basic reporting feature through Jasper Reports. This basic feature is reconfigured and extended to the needs of the current system. Reporting is associated with all processes and data transactions. Reports related to results also contain graphical elements to emphasize on the meaningful aspects of the analyzed data. Reports can also be collected for information related to the management of the system and at the project level. These management reports involves summary of progress, intermediate results achieved, specific notes by the analyst or data handler or system alerts related to infrastructure and system errors.

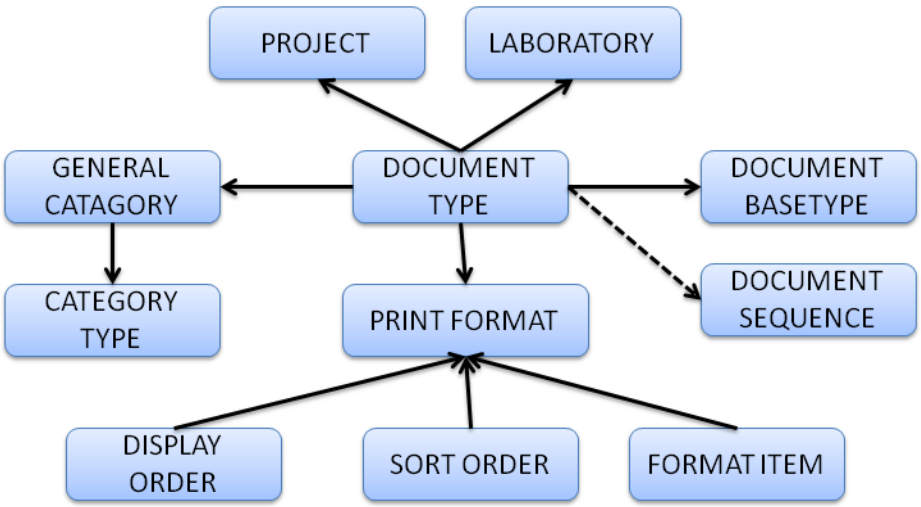


Figure 21 Documentation structure for dynamic reporting

The document type is the general starting class access to which is verified from the project and laboratory of the user. It identifies the reporting type through the type of category of the general category entry that initiated the request. The category type identifies the report as a summary report, results etc. The document base type identifies the starting point of the document and the document sequence specifies the order of the pages. This is important in reports with multiple pages from different sources. Once all document source and meta information are collected, the print format identifies how data is rendered for printing. This is achieved through identification of the standard format items included in the document and their display and sort orders. This format information and the actual print data are merged to create the report and are rendered for printing.

Format Item	Description
@*Page@	Current page number
@*PageCount@	Total number of pages
@*MultiPageInfo@	'Page x of y' for multiple page documents
@*CopyInfo@	If is a copy of document the 'Duplicate' is printed
@*ReportName@	Name of the report
@*Header@	Full header with user/project/laboratory name
@*CurrentDate@	Printing date
@*CurrentDateTime@	Printing date and time

Table 2 Examples of the format items (adapted from [85])

Within AD2BioDB reports are related to project management reporting, tracking and system reporting. Project management reports include status, progress and deliverable. Data movement, issues related to tasks and resource usage are tracked within the system and are dynamically reported. System reporting is related to the management of the application i.e. access logs, application errors, system level user requests, etc.

This dynamic reporting depends on the definition of supervision rules defined through the accounts definition. The basic project management rules are predefined within the system but can be modified and improved. Through the complex accounting structure, project supervision is dynamically implemented. An example of the accounting and reporting can be understood in the analysis scenario, where data is collected for analysis and the results are added into the system. For the automated supervision, data usage needs to be tracked; import of analysis data takes place; results needs to be validated and upon acceptance of the result the project progress needs to be updated. All these tracking are automatically achieved through the predefined supervision rules. Thus supervisors can view reports and need not worry about the availability of information to others.

4.2 SeqPipe

SeqPipe is a flexible system for management of dynamic workflows in a distributed environment. The software environment consists of an application server, data server and several analysis servers. The need for flexibility is related to tools, data formats, workflow creation, execution of jobs in analysis servers and changes in the availability of analysis servers. Users should be able to access the system through web clients or programmatic access. The ability of the system to be plugged into other system is essential as such a system is mostly related to larger processes in a research facility.

4.2.1 System architecture

SeqPipe uses the same platform structure as that of AD2BioDB with the exception that it is not database driven. This implies client-server architecture with model view controller design. The dynamic application data are stored in synchronized files with concurrency support implemented through multithread supported wrapper classes. The web client uses ZK for presentation of the application visual components. The platform supports generic classes for connecting to a particular analysis server using server specific data. Such connections are established through secure SSH channel provided by the JSch package, thus providing security over the network.

In the pluggable environment the application needs an interface class to be extended with the description of the data source. In this way the management component of the system is utilized instead of the SeqPipe management. Further, through the data source function the application data can be linked to the system application data store. At the time of the plugin deployment for an individual system the SeqPipe application data tables need to be defined.

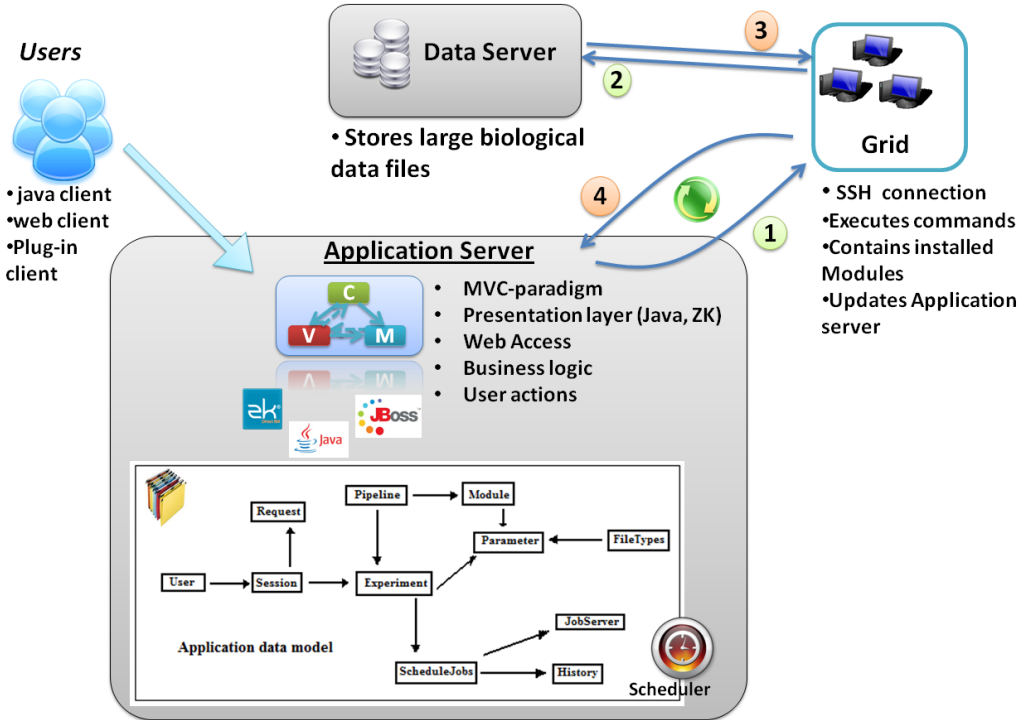


Figure 22 SeqPipe architecture with the application data model. The numbered arrows indicate the steps in the execution of a command request. On receiving the request from the application server the data is collected from the data server and is processed. Then the results are placed in the data server and the application server is updated of the status.

The implementation of the functionality of the working data model is described as:

i) New Parameter and File Type creation: Parameters consist of the name, value and parameter type. Parameter also contains information of its association with the relevant Module or workflow. If the parameter type is that of file then the corresponding file handler, carrying read write operation, needs to be defined in the FileType class.

ii) New module creation: Module is the equivalent of command for any analysis tool and is defined in the Module class, by the versioned name, command to be executed, parameter default values and server ids that can execute the module. The command along with the parameters forms the complete executable statement.

iii) New analysis server addition: New servers for analysis are prepared by the system administrator. The server connection information is then added to the server class. This is followed by updating the modules that are supported by the server.

iv) New workflow creation and preparation for execution: A workflow is defined in the pipeline class. The workflow name is automatically associated with a version in order to track changes made to the workflow. Further the workflow consists of one or more modules and workflow parameter values and batch parameters. The addition of modules into the pipeline is validated by a continuity of the parameters. For the input parameters of an intermediate module, at least one output parameter of the previous module should be included.

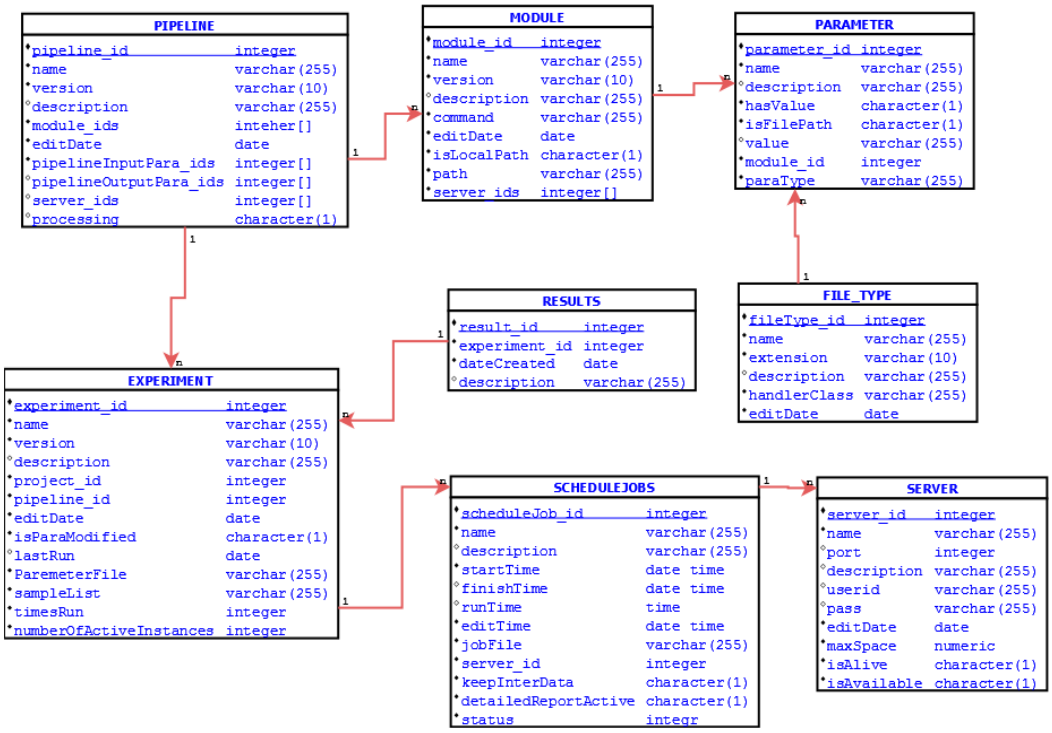


Figure 23 Scheme of SeqPipe data flow

While the pipeline class is involved in maintaining the pipeline, functionality is provided by the experiment class such that for a particular pipeline multiple experiments can be carried out simultaneously. Experiment class assigns non-default values to the parameters and also needs project name in its definition. Date of last execution and number of times the experiment has been conducted gives a meta-assessment of its relevance.

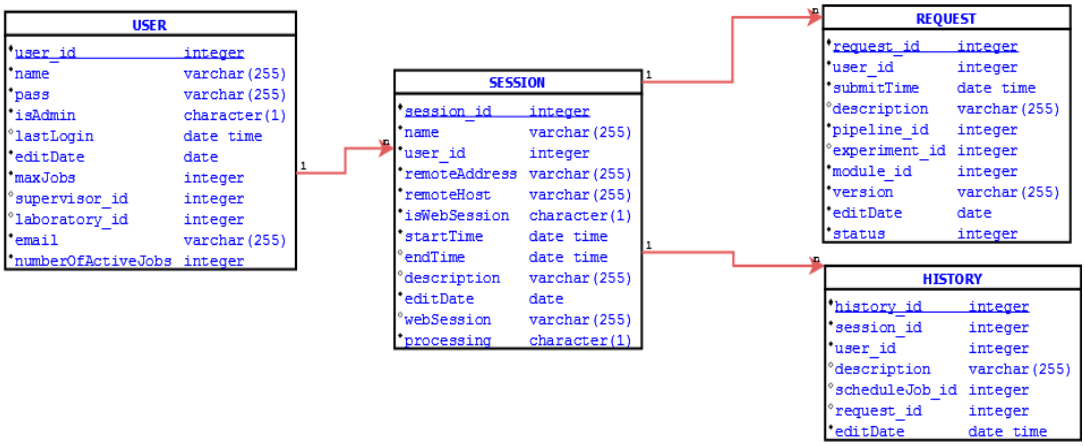


Figure 24 Administrative data flow within SeqPipe.

4.2.2 Distributed analysis server

Each analysis server needs a job client to be installed before it is made available in the system. The job client is a SSH server with which the application server interacts with. Modules needs to be explicitly installed and made available for execution and is a part of the application server setup. The job client also contains a ssh client for connecting to the data server. Through this connection the data are collected for analysis. Once the data is completely transferred into the analysis server, execution of the workflow begins. Workflow is represented by a set of commands executed sequentially and the application server is updated of the status as each command statement.

4.2.3 Scheduling and running workflows

The user configures an experiment and then schedules the execution. At the time of execution of the job, availability of servers that can execute the pipeline is checked and assignment is done in a greedy selection among the available servers. If all the servers are busy then the server with least number of queued jobs, is selected. For the execution of the workflow commands are sent one at a time and the thread waits for the response. The responses include completion or error. Upon completion of the workflow run, results are then transferred to the data server and the application server receives a summary of the results. This information is stored and processed for updating the user.

5 System configuration for genome sequencing technology and Case study

Although the system is pre-configured with management section, technological configuration has to be done for individual technologies. For any new technology the following are the minimal set of information needs to be added:

i) Technological platform that is the information regarding the instrument used for the data generation.
ii) Description of the data formats used for integration into the system and the tool for the execution of read/write functions.
iii) Other details related to the new technology needs to be created in new database tables and linked to the default technology table.

All related classes and wrappers are then prepared for a complete support of the new technology within the system. For analysis the relevant tools and commands are incorporated into the system and can be extended to the creation of pipelines and allocation of analysis servers. The definition of workflows can be carried out by the system administrator only. This is because the analysis servers need to be configured for supporting the new workflows. The system administrator needs to prepare all windows related to the new technology and the document structure for reporting. The preconfigured scheduler service should be available with the functional application server.

After the required technologies are defined, the system administrator can then create the new project that uses the new technology and define the organizational structure. The supervision accounts needs to be defined next. Through these accounts the complex process of reporting and tracking are identified. AD2BioDB provides a default set of rules for supervision and in most projects the same accounting structure may be followed. For each project, a project admin and project user roles are automatically created while further roles needs to be specified later. After the creation of roles the users are added for the project. The collaborator information and relation rules are also established manually. Collaborators can include other research centers or individuals. Now data can be imported into the system. Data import can be carried out by the Supervisor or Data handler roles. For the completion of data import process, metadata about the subjects are also added along with data file location to the system.

With the completion of the setup process for a new project users can log into the system. Login process includes first the connection part where the user credentials are verified. Here changes in the application server and database can be made by clicking on the Server information area. Thus by the use of the same installation various running implementations can be accessed. The second part of the login process is the definition of the purpose by the selection of the information in the defaults tab. As this step is completed the systems is aware of the resources needed for the user and the application dashboard is made available. The user can then start working by selecting the menu items.

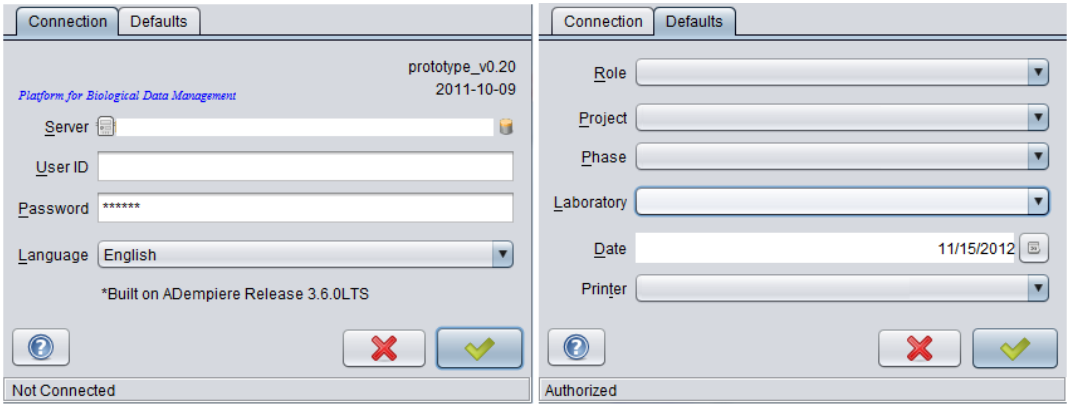


Figure 25 User Login page indicating controlled access. First the user's credentials are verified in the connection tab and the purpose is defined through the defaults tab. Controlled access to the application is then provided to the user.

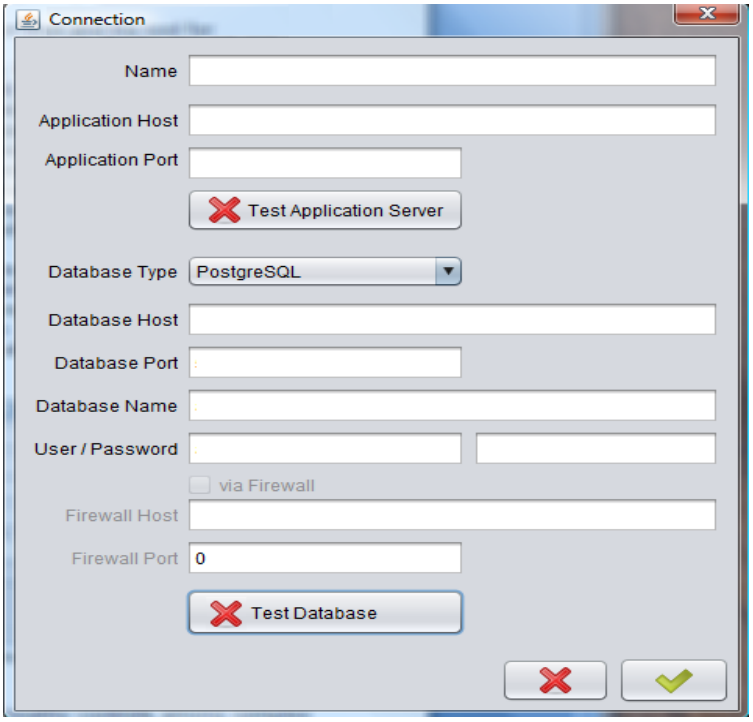


Figure 26 Connection window is used to define the application server address and the database address used for running AD2BioDB

HYPERGENES project [86] was centred on the objective of Integrating biological data and processes with Hypertension as the disease model. The HYPERGENES project focuses on the definition of a comprehensive genetic epidemiological model of complex traits like Essential Hypertension (EH) and intermediate phenotypes of

hypertension such as Target Organ Damage (TOD). It is a collaborator project involving many research institutions with each having a specific set of deliverables. Thus the setting up of the project within AD2bioDB involves the collaborator project structure, clinical phenotype hierarchy and technological accounts.

In a usage scenario, the sequencing technology related data is discussed here. Sequencing data was generated by Dr. Cristina Barlassina, Dr. Francesca D'Avila and Dr. Daniele Braga while the analysis was carried by Dr. Sara Lupoli and Dr. Matteo Barcella. 44 hypertensive cases and 48 healthy controls among samples collected by the HYPERGENES consortium and genome-wide genotyped with the Illumina 1M array were selected for a fine mapping, through deep sequencing, of a genomic region identified in the genome-wide association study. The rationale involves the results of a Genome wide association study (GWA) generally highlight candidate regions of the genome but only occasionally point to "causal" variants and these variants have to be looked for with other approaches and target sequencing with next generation sequencing technology together with target enrichment strategies are powerful methods to deal with this issue. Goals of a sequencing approach are to describe the entire nucleotide sequence underlying the candidate regions, with all single nucleotide polymorphisms (SNPs) and regulatory domains in order to identify the "functional" SNPs. In this way we should be able to understand the real meaning of the significant association results of the GWA. All samples collected were self-reported and empirically confirmed using GWAS data of continental Italy ancestry.

Target re-sequencing pipeline was run on the raw data. The following steps composed the sequencing pipeline:

- Paired-end raw reads were checked for their quality using FastQC [87] and trimming 3'-end was performed by PrinSeq [88] if a negative summary statistics was obtained. This produces as output the sam-format file which contains all information about reads like position in the reference genome, sequence length, base quality score, sequences quality score, mate pair, insert size and many others information which describe the nature of the data.In particular, starting from the right-end of each read we trimmed bases with a quality score below 15, keeping only reads with a minimum average base quality of 28 and a minimum read length of 45.
- Reads were aligned to the human reference genome (hg19, UCSC assembly, February 2009) using BWA [56] a light-weighted tool that aligns relatively short sequences (queries) to a sequence database (target). SAMtools ([58], [89]), Picard [90] and Genome Analysis Toolkit (GATK) [57] were used to handle the reads and for post-alignment quality control checks. Post alignment quality control allows creating a cleaned bam file, which can be used as input for variant discovery. The following were performed :
    1) Local realignment is necessary if there is the presence of an indel (insertion/deletion) in the individual's genome compared to the reference genome and it is an efficient and effective way to rule out false SNPs caused by nearby INDELs. This procedure is made using a

multi-reads approach. The wrong alignments lead to recurrent mismatches, which are likely to deceive most site-independent SNP callers into calling false SNPs. Local realignment allows transforming regions with misalignments due to indels into clean reads containing a consensus indel suitable for standard variant discovery.

2) Marking duplicate is a common practice to get rid of perfect duplicates which can affect variant discovery process. Reads duplicates are mainly produced by PCR cycles, which creates several copies of the same read. This marking is necessary to avoid a mis-call in a variant discovery approach.
3) The base quality score recalibration serves to handle base quality scores that are closer to their actual probability of mismatching the reference genome. The recalibration tool attempts to correct for variation in quality with machine cycle and sequence context, and by doing so provides not only more accurate quality scores but also more widely dispersed ones. It permits to level out bias coming from different sequencing techniques and protocols. This process analyzes several covariates like reported quality score, sequencing chemistry effect, position within the read, that are used to recalibrate base quality score in reads described in a bam file. Hence, depending on sequence context, a base quality score can be affected by a bias which can be corrected by this process.

- Multi-samples variant call was performed by GATK applying the default parameters. The algorithm used is based upon the bayesian theory and estimates genotyping likelihood for each allele combination. In the algorithm are counted only the bases with a base quality score above 17 (range 2 - 40) and the reads with a mapping quality score above 20 (range 0 - 60). The software also emits a score that describes the quality of the variant site. The higher is the value of this score, the higher is the probability that the variant is true. The input file of this module is the bam file (samtools binary format) that is subjected to several quality control processes, which will be described in the quality control step post alignment. The output is a variant call file (.vcf format), which contains all information about genomic variants present in the sample analyzed. Quality filters were applied to variant call; in particular sites with strand inconsistency and clusters of variants suggestive of read misalignment were removed. Only variants with high SNP quality score were kept.

For novel SNPs, not previously annotated in dbSNP135, in order to minimize the false positive, additional filters were used. In particular we set the minimum genotype score per sample at 50 and the number of reads covering the variant at 10 reads per sample. Annotation of the good quality variants was performed using different software and EBI, NCBI and Ensemble databases.

6 Conclusion and Discussion

We developed AD2BioDB as a prototype, as an endeavor to solve the complexities related to clinical genomics data management. Through AD2BioDB we have created a software system that can be robust enough to encompass the requirements of the domain on one platform yet being flexible enough to adjust to the volatile and evolving technologies. The distributed nature of implementation helps the system in being immune towards the infrastructural changes yet the system is compact enough to be implemented in a single desktop environment. AD2BioDB is platform independent and is configured for working on Windows, Linux and Mac OS that contains Java and a database management system. The increasing demands for security has been implemented through the use of role based controlled access and separation of the patient demographics and other metadata related to the patient. The management of users and projects with a dynamic reporting capability facilitates AD2BioDB's use in any research center. Data management according to the technological classification, visualization through linked visualizer and the use of public knowledge in the same system provides a complete research experience. An important aspect of AD2BioDB is the collaboratory research environment being provided as collaborators can be continuously informed and updated about the progress of the project. Important findings and observations can be made available among all members quickly. Further an assessment of the performances can be made periodically about the project status.

AD2BioDB, following the open source direction of its framework ADempiere, has a GNU open source license. Once AD2BioDB reaches the deployment phase the source code will also be available for the community. This would help in greater improvement of the software's features, through community participation. In the development of AD2BioDB a subversion system [91] has been used to keep track of the development path.

As a sub domain for the system, through SeqPipe we have tried to provide solution to another challenging aspect concerned with the analysis of next generation sequencing data. SeqPipe workflow creation and execution can be carried even in laboratories with a modest number of desktop computers connected in a network. The system can also be used in bigger laboratories containing clustered servers and hence has high scalability. The dynamic nature and simplicity in the creation of workflow makes SeqPipe an easy workflow creation tool. For the execution of workflows, SeqPipe has control over each command being executed and can implement flow control operation on the running workflow. Meta information maintained within the system enforces reproducibility. Within AD2BioDB, SeqPipe is complete integrated, to provide the key functionality of the maintenance and execution of analysis workflows.

Understanding the needs of the scientific community for a minimal coding environment in the use of tools, AD2BioDB framework provides the dynamic feature of application dictionary. Through this new representations of the data in

widows and new questioners can be created without any code being written. This feature is of great help to scientists who need to collect information and conduct surveys. The participants have very limited temporary access to the system only for the predefined purpose. Thus AD2BioDB provides a secure automated data collection framework as well and the data can be automatically processed. AD2BioDB, as a research management platform, will be helpful to all research laboratories when configured according to their needs. It provides a single solution to the automated analysis and data storage. Project planning and management in AD2BioDB provides well managed environment for doing research with automated supervision through dynamic reporting. Further the clinical participation in genomic research directly benefits through this platform.

7 Limitations and future directions

As a prototype AD2BioDB has many limitations before it can be tried at the deployment phase. Although the system can be configured for all high throughput data, presently AD2BioDB is only configured for sequencing data. This limitation will not be removed at the prototype level and other technologies will be configured only when the prototype upgrades to the deployment phase.

AD2BioDB presently provides public annotation data through the UCSC genome database. Although the integration system have been formulated, but complete implementation with mapping of the major data sources is yet to be achieved. A related and unavoidable limit is automatically imposed on the query system. This is at a basic level of development as it follows the complete integration of the public annotation data sources.

For the use of AD2BioDB in the clinical domain requires a greater implementation of the standards and protection of the patient demographic information. With the perspective of a genomics laboratory it can be argued that the patient information can be provided in a codified format such that only information related to the study is available. Detailed information about the patient can be maintained with the clinician involved. Thus the implementation of standards will be carried at the deployment phase of the system development cycle.

Although not a limitation for the prototype, but definitely feature to be provided in future is the incorporation of complete LIMS functionality for AD2BioDB. This mainly relates to the integration of laboratory instruments to provide an automated research environment with the continuity of in-silico data being generated by the devices till the analysis and generation of meaningful conclusions.

With Ad2BioDB achieving a complete set of features and functionality is should be able to provide a decision support system for the clinicians. Although not in the present scope of implementations, but a system for electronic data capture for patients can be implemented if AD2BioDB is to be implemented in the clinical setting.

For SeqPipe, the major limitation is the lack of a graphical workflow creation tool. For this reason the workflows creation involves many steps of addition of data into the system. Further the transfer of data between data server and the analysis server is slow. This limitation is subject to research in informatics disciplines and new protocols are being developed for the transfer of large data over the network.

8 Bibliography

[1] Schadt, Eric E., Björkegren, Johan L M; **NEW: network-enabled wisdom in biology, medicine, and health care**; Science translational medicine; 2012 4: 115; DOI: 10.1126/scitranslmed.3002132

[2] International Human Genome Sequencing Consortium. **Finishing the euchromatic sequence of the human genome**. Nature, 2004, 431(7011):931-945.

[3] Frederick L. Kiechle, Xinbo Zhang, Carol A. Holland-Staley; **The -omics Era and Its Impact;** Arch Pathol Lab Med; 2004 128;

[4] Jason R Swedlow, Gianluigi Zanetti & Christoph Best; **Channeling the data deluge;** Nature Methods; 2011 8(6);

[5] Genomics and World Health: Report of the Advisory Committee on Health research, Geneva, WHO (2002)

[6] Margaret Mary Brosnahan, Samantha A. Brooks,* and Douglas F. Antczak . **Equine clinical genomics: A clinician's primer.** Equine vet. J. 2010, 42 (7) 658-670

[7] M. Ronaghi, S. Karamohamed, B. Pettersson, M. Uhlen, and P. Nyren. **Real-time DNA sequencing using detection of pyrophosphate release**. Analytical Biochemistry, 242(1):84-9, 1996.

[8] M. Margulies, M. Egholm, W. E. Altman, S. Attiya, J. S. Bader, L. A. Bemben, J. Berka, M. S. Braverman, Y. J. Chen, Z. Chen, S. B. Dewell, L. Du, J. M. Fierro, X. V. Gomes, B. C. Godwin, W. He, S. Helgesen, C. H. Ho, et al. **Genome sequencing in microfabricated high-density picolitre reactors**. Nature, 437(7057):376{80, 2005.

[9] D. R. Bentley, S. Balasubramanian, H. P. Swerdlow, G. P. Smith, J. Milton, C. G. Brown, K. P. Hall, D. J. Evers, C. L. Barnes, H. R. Bignell, J. M. Boutell, J. Bryant, R. J. Carter, R. Keira Cheetham, A. J. Cox, D. J. Ellis, et al. **Accurate whole human genome sequencing using reversible terminator chemistry**. Nature, 2008, 456(7218):53-9.

[10] J. Shendure, G. J. Porreca, N. B. Reppas, X. Lin, J. P. McCutcheon, A. M. Rosenbaum, M. D. Wang, K. Zhang, R. D. Mitra, and G. M. Church. **Accurate multiplex polony sequencing of an evolved bacterial genome**. Science, 2005, 309(5741):1728-32,

[11] T. D. Harris, P. R. Buzby, H. Babcock, E. Beer, J. Bowers, I. Braslavsky, M. Causey, J. Colonell, J. Dimeo, J. W. Efcavitch, E. Giladi, J. Gill, J. Healy, M. Jarosz, D. Lapen, K. Moulton, S. R. Quake, K. Steinmann, E. Thayer, A. Tyurina, R. Ward, H. Weiss, and Z. Xie. **Single-molecule DNA sequencing of a viral genome**. Science, 320(5872):106-9, 2008.

[12] J. Korlach, P. J. Marks, R. L. Cicero, J. J. Gray, D. L. Murphy, D. B. Roitman, T. T. Pham, G. A. Otto, M. Foquet, and S. W. Turner. **Selective**

aluminum passivation for targeted immobilization of single DNA polymerase molecules in zero-mode waveguide nanostructures. Proceedings of the National Academy of Sciences USA, 105(4):1176-81, 2008

[13] W. J. Ansorge. **Next-generation DNA sequencing techniques**. Nature Biotechnology, 25(4):195-203, 2009

[14] M. Kircher and J. Kelso. **High-throughput DNA sequencing-concepts and limitations**. BioEssays, 32(6):524-36, 2010.

[15] Margulies, M. *et al.* **Genome sequencing in microfabricated high-density picolitre reactors**. *Nature* 2005, **437**, 376–380.

[16] Bentley, D. R. **Whole-genome re-sequencing**. Curr. Opin. Genet. Dev. 2006, **16**, 545–552.

[17] Maskos, U. & Southern, E. M. **Oligonucleotide hybridizations on glass supports: a novel linker for oligonucleotide synthesis and hybridization properties of oligonucleotides synthesised in situ**. Nucleic Acids Res. 1992, 20: 1679–1684.

[18] Pease, A. C. et al. **Light-generated oligonucleotide arrays for rapid DNA sequence analysis**. Proc. Natl Acad. Sci. USA. 1994, 91: 5022–26.

[19] Hughes, T. R. et al. **Expression profiling using microarrays fabricated by an ink-jet oligonucleotide synthesizer**. Nature Biotechnol. 2001, 19: 342–47.

[20] David Gresham, Maitreya J. Dunham and David Botstein. **Comparing whole genomes using DNA microarrays.** Nature Genetics Review. 2008, 9: 291-302.

[21] Amnon Shabo (Shvo). **Clinical genomics data standards for pharmacogenetics and pharmacogenomics.** Pharmacogenomics. 2006, 7(2): 247-253, DOI 10.2217/14622416.7.2.247.

[22] Craig, J. **Complex diseases: Research and applications.** *Nature Education.* 2008, 1(1)

[23] Jahromi MM, Eisenbarth GS. Cellular and molecular pathogenesis of type 1A diabetes. Cell Mol Life Sci. 2007;64:865–72

[24] John Bell. **Predicting disease using genomics.** Nature. 2004, 429: 453-56.

[25] Detmer DE, Lumpkin JR, Williamson JJ. **Defining the Medical Subspeciality of clinical Informatics**. J Am Med Inform Assoc 2009; 16:167-8

[26] Amnon Shabo (Shvo). The implementation of electronic health records for personalized medicine. Personalized Medicine. 2005, 2(3): 251-28.

[27] Wolf MS el. al. **Managing incidental findings and research results in genomic research involving biobanks and archived data sets**. Genetics in Medicine. 2012, 14(4):361-84.

[28] **International Classification of Diseases:** www.who.int/classifications/icd/en/

[29] **Systematized Nomenclature of Medicine:** www.snomed.org/

[30] **Logical observation identifiers names and codes:** www.regenstrief.org/loinc/

[31] **Health Level seven (HL7):** www.hl7.org/

[32] **Digital Imaging and Communications in Medicine, Standard Specification:** http://medical.nema.org/dicom.html

[33] Committee European Normalisation, CEN/TC 251 Health Informatics Technical Committee. **Health Informatics – Electronic healthcare record communication – Part 1: extended architecture. ENV13606-1:** www.openehr.org/standards/t_cen.htm

[34] Rocca-Serra P. et. al. **ISA software suite: supporting standards-compliant experimental annotation and enabling curation at the community level**. BMC Bioinformatics. 2010, 26(18): 2354-56

[35] Tim F. Rayner et al. **A simple spreadsheet-based, MIAME-supportive format for microarray data: MAGE-TAB. BMC Bioinformatics**. 2006, 7: 489

[36] **Clinical Data Interchange Standards Consortium**: http://www.cdisc.org/

[37] **Study Data Tabulation Model:** http://www.cdisc.org/sdtm/

[38] **Sequence Read Archive XML format(SRA-XML):** http://www.ebi.ac.uk/ena/about/sra_preparing_metadata

[39] **International Nucleotide Sequence Database Collaboration (INSDC):** www.insdc.org

[40] Kathy L. Hudson, **Genomics, Health Care, and Society**. N Engl J Med. 2011, 365(11): 1033-41

[41] Ellen Wright Clayton. **Ethical, Legal, and Social Implications of Genomic Medicine**. N Engl J Med. 2003, 349(6): 562-9

[42] Martin S. Kohn, Hot Topics Work Group, Management Engineering and Process Improvement Task Force, HIMSS. **Rapid Change in Healthcare Organizations**. 2007.

[43] Kerlavage A, et al. **Data management and analysis for high throughput DNA sequencing projects**. IEEE Eng Med Biol. 1995 14:710–717

[44] Eric E. Schadt, Michael D. Linderman, Jon Sorenson, Lawrence Lee, Garry P. Nolan. **Computational solutions to large-scale data management and analysis.** Nature Genetics reviews. 2010, 11(9):647-57

[45] Laura DeFrancesco, Nidhi Subbaraman. **Sequencing firms eye pathology labs as next big market opportunity**. Nature Biotechnology. 2011, 29(5): 379-80

[46] Sansone S, et al. **Toward interoperable bioscience data**. Nature Genetics. 2012, 44(2):121-6

[47] **I2B2**: www.i2b2.org

[48] **caBIG**: http://cabig.cancer.gov/

[49] Nuzzo A., Riva A. **Genephony: a knowledge management tool for genomewide research**, BMC Bioinformatics 2009, 10:278 doi:10.1186/1471-2105-10-278

[50] Adida B., Kohane I.S., **GenePING: secure, scalable management of personal genomic data**, BMC Genomics 2006, 7:93 doi:10.1186/1471-2164-7-93

[51] Chiang GT, Clapham P, Qi G, Sale K, Coates G., **Implementing a genomic data management system using iRODS in the Wellcome Trust Sanger Institute**. BMC Bioinformatics. 2011, 12:361.

[52] Mavromatis K, Chu K, Ivanova N, Hooper SD, Markowitz VM, et al. **Gene Context Analysis in the Integrated Microbial Genomes (IMG) Data Management System**. PLoS ONE. 2009, 4(11): e7979. doi:10.1371/journal.pone.0007979

[53] **1000 Genome Project**: http://www.1000genomes.org/

[54] **TCGA**: http://cancergenome.nih.gov/

[55] **10K Genome Project**: http://www.genome10k.org/

[56] Li H. and Durbin R. **Fast and accurate short read alignment with Burrows-Wheeler transform.** Bioinformatics, 25: 1754-60, 2009

[57] **GatK:** http://www.broadinstitute.org/gatk/

[58] **SAM format and SAMtools**. Bioinformatics, 25: 2078-9, 2009

[59] **Galaxy**: http://galaxy.psu.edu/

[60] Hull D, Wolstencroft K, Stevens R, Goble C, Pocock MR, Li P, Oinn T. **Taverna: a tool for building and running workflows of services.** Nucleic Acids Res. 2006, 1:34(Web Server issue):W729-732.

[61] Monya Baker, **Next-generation sequencing: adjusting to data overload nature methods**, 2010, 7(7):495

[62] **Who Genomic benefits:** http://www.who.int/trade/distance_learning/gpgh/gpgh5/en/index3.html

[63] W. Sujansky, "**Heterogeneous database integration in biomedicine**." J Biomed Inform, 2001, 34(4): 285-298

[64] B. Louie, P. Mork, F. Martin-Sanchez, A. Halevy, and P. Tarczy Hornoch, "**Data integration and genomic medicine.**" J Biomed Inform, 2006 40(1): 5-16,.

[65] M. Mesiti, E. Jiminez-Ruiz, I. Sanz, R. Berlanga-Llavori, P. Perlasca, G. Valentini, and D. Manset, "**Xml-based approaches for the integration of heterogeneous bio-molecular data**." BMC Bioinformatics, vol. 10 Suppl 12, p. S7, 2009.

[66] Cantor M., Dubchak I., Gordon D., Wang T. **Visualizing genomes: techniques and challenges**. Nature Method Supplement 2010 7(3)

[67] Skinner, Mitchell E., Uzilov,Andrew V., Stein,Lincoln D., MungallChristopher J., Holmes Ian H.; JBrowse: **A next-generation genome browser**; Genome Res. 2009; 19: 1630-1638 , doi: 10.1101/gr.094607.109

[68] Shah S.P., He D.Y., Sawkins J.N., Druce J.C., Quon G., Lett D., Zheng G.X.Y., Xu T., Ouellette B.F.F., **Pegasys: software for executing and integrating analyses of biological sequences**, BMC Bioinformatics 2004, 5:40

[69] **Patterns**: http://en.wikipedia.org/wiki/Architectural_pattern

[70] Fowler M, **Patterns of enterprise application architecture**, Addison-Wesley Professional,2003

[71] **JBoss:** http://www.jboss.org/

[72] **Java:** http://java.com

[73] **J2EE:** http://java.sun.com/j2ee/

[74] http://www.oracle.com/technetwork/java/javaee/ejb/index.html

[75] **ADempiere:** http://www.adempiere.com/ADempiere_ERP

[76] **GPL:** http://www.gnu.org/licenses/gpl.html

[77] Staeuble M., Schumacher J., **Zk Developer's Guide**, Packt Publishing, 2007

[78] **Ajax:** http://code.google.com/edu/ajax/tutorials/ajax-tutorial.html

[79] **RIA:** http://www.w3.org/TR/wai-aria/

[80] **UCSC:** http://genome.ucsc.edu/

[81] **NCBI:** http://www.ncbi.nlm.nih.gov/

[82] **Ensembl:** http://www.ensemblgenomes.org/

[83] **dbSNP:** http://www.ncbi.nlm.nih.gov/projects/SNP/

[84] **Perl:** http://www.perl.org/

[85] Pelgrim, **ADempiere Functional Design:** http://adempiere.org/schema/ADempiereFunctionalbasics.html

[86] **HYPERGENES:** http://www.hypergenes.eu/

[87] **Fastqc**: http://www.bioinformatics.babraham.ac.uk/projects/fastqc/

[88] **Prinseq:** http://prinseq.sourceforge.net/

[89] Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R. and 1000 Genome Project Data Processing Subgroup (2009). **The Sequence alignment/map**

[90] **Picard**: http://picard.sourceforge.net/

[91] Pilato C.M, Collins-Sussman B, Fitzpatrick B.W, **Version control with subversion**, O'Reilly 2004.