**06-HG-101\* New computational and statistical methods for the analysis of large data sets from next-generation sequencing technologies. (High priority grant)s**

The introduction of new methods for DNA sequencing has opened new avenues, including large-scale sequencing studies, metagenomics, transcriptomics, genetic network analysis, and determination of the relationship of sequence variation and phenotypes to disease, to address heretofore unapproachable problems in biomedical research. However, since the large amounts (terabases) of data generated overwhelm existing computational resources and analytic methods, urgent action is needed to enable the translation of this rich new source of genomic information into medical benefit. Contact: Dr. Lisa Brooks, 301 496-7531, [brooksl@mail.nih.gov](mailto:brooksl@mail.nih.gov)

(*See end of document for Research Plan structure and format*)

**Specific aims**

NextGen sequencing technologies are fast approaching the ‘$1,000 genome’ target (1), an effort to develop sequencing technologies that are inexpensive and efficient enough for biomedical research and health care labs to routinely sequence entire genomes. A $5,000 genome will likely be attained in mid-2009 by Comparative Genomics while other NextGen industry players are following suit by steadily increasing run yield and reducing cost per Mbase. A new paradigm is emerging of the correlated and rapid analysis of individual genomic variation, methylation, histone-binding, expression analysis and other genome-wide factors that may begin to unlock the secrets of the cell (2) and create new avenues for clinical diagnostics.

Bioinformatics infrastructure – hardware, software and personnel – is the bottleneck in the development of this new paradigm (2, 3). Costly investments of high performance computing resources and skilled personnel are required to develop bioinformatics algorithms, integrate diverse biological data sources and devise ways to visualize and interpret these huge data sets. Most biomedical research and health care labs are unable to provide even the minimum of these requirements.

The specific aims of this study are designed to meet these requirements by providing a tool for managing NextGen sequencing projects, from short read generation to bioinformatics analysis to data visualization. These aims are as follows:

1. **Assembly Methodology and Pipeline**
   1. Evaluate all current NextGen assemblers/aligners
   2. Provide an optimized NextGen assembly pipeline
   3. Enable pipeline customization by end user
2. **SNP Annotation and Expression Analysis**
   1. Defined quality control/assurance algorithms for SNP verification
   2. Develop a SNP annotation pipeline
   3. Integrate with current expression analysis packages
   4. Develop new expression analysis algorithms
3. **Reporting and Visualization** 
   1. Design data filters based on user requirements
   2. Provide genome-level view of data with integrated external databases
   3. Integrate with downstream analysis applications (Cytoscape, Genespring, R, etc.)
4. **Development of Comprehensive Software** 
   1. Establish software design plan based on requirements in Aims 1-3
   2. Implement user-centric development process to ensure usability
   3. Execute thorough post-release user testing at each project milestone

The aims do not include the following:

Workflows including biological web databases or applications that do not provide Web services

Workflows including Windows-only applications

LIMS (laboratory instrumentation management systems) capabilities

Comparative genome visualization

**Background and Significance**

**NextGen Technologies**

The current mainstream NextGen platforms produce millions of short (50bp – 400bp) sequence reads. Each of the three main platforms, namely, Illumina/Solexa (4), Roche/454 (5) and ABI/SOLiD (6) have their own inherent problems, including significant sequencing error rates and systematic errors. Despite an initial capital outlay of $600 000 to $1.35 million, the cost per base is substantially lower than for Sanger sequencing. Large sequencing organizations such as genome centers, academic core facilities and commercial contract-sequencing enterprises across the globe have already adopted this NextGen technology (Figure 2) and smaller labs and molecular diagnostics facilities participating in growing numbers.

Prior to sequencing, NextGen sample preparation varies considerably but usually involves multiple steps taking 2–4 days to complete,depending on the platform. ‘Barcodes’ – unique identifier sequences added to reads – can also be used to analyze multiple samples within the same separate flow-cell lanesor compartments. These barcodes are ligated to individual samples which are then pooled and sequenced and later separated out based on their barcode.Barcode-based multiplexing and other incremental innovations in process streamlining, automation and chemistryrefinements will continue to reduce costs and sequencing errors. The recent rollout of paired-end reads (a.k.a. mate-paired reads) – short reads that flank a region of known length in the sample sequences - by all of the major platforms has provided a major advance in de novo assembly and the correction of reference alignment errors by eliminating alignments that do not match the size of gap between the paired reads (7). There is also a strong need for flexible and effective targeted capture methods for isolating reduced genomic subsets – such as genomic regions or exons of candidate genes – implicated in disease prior to NextGen sequencing of multiple individuals. Different approaches have already shown proof-of-concept, such as 10,000 enrichment by hybridizing biotinylated BACs (bacterial artificial chromosomes) with targeted segments of genomic DNA (8), microarray-based enrichment of several kilobase-sized human genomic regions (9) and multiplex PCR amplification of 170 exons (10). **Two commercial capture methods are currently available: Nimblegen and Agilent.**



Figure 2. NextGen Sequencing Technology Roadmap

**3G Technologies**

Third generation sequencing technologies are being developed to sequence single DNA molecules faster and cheaper with streamlined samplepreparation. Real-time sequencing by synthesis is being developedby VisiGen (<http://www.visigenbio.com>) and Pacific Biosciences(<http://www.pacificbiosciences.com>). Pacific Biosciences is due to launch commercially in 2010 and has a mean DNA synthesisrate of approximately 4 bases per second, with a maximum read length of 4,000 bp.Also in development is sequencing based on sensing the bases of DNA molecules passed through nanopores (~5 nmin diameter). Different methods are being tested to create nanopores, including inorganicmembranes (solid-state nanopores), genetically engineered protein channelsby Oxford Nanopore Technologies(<http://www.nanoporetech.com>), polymer-based nanofluidicchannels, and a combinationof nanopores with sequencing by hybridization by NABsys (<http://www.nabsys.com>). As these technologies develop, they will present new bioinformatics problems to be solved and greater data infrastructure demands.

**NextGen Applications**

A growing variety of molecular methods has enabled the investigation of a broad range of biological phenomena by high-throughput DNA sequencing, including genetic variation, RNA expression, protein-DNA interactions and chromosome conformation (7). Chromatin immuno-precipitation (ChIP) is used to investigate protein-DNA interactions, which play a key part in regulating gene expression and controlling the availability of DNA for transcription and replication. In the technique, DNA chemically cross-linked to associated proteins is fragmented and transcription factor-specific antibodies are used to immunoprecipitate selected protein-DNA complexes. The DNA is then processed by NextGen sequencing (ChIP-Seq). Expression profiling (a.k.a. RNA-seq or transcriptome analysis) is another popular NextGen application that has been demonstrated as robust and sensitive in comparison to five microarray platforms (11). Furthermore, microarrays cannot detect antisense transcription, which was found in 51% of all genes. Transcriptomes for mouse brain, liver and skeletal muscle were mapped by NextGen deep sequencing (12), providing a digital measure of the presence and prevalence of transcripts from known and previously unknown genes. RNA standards were used to quantify transcript prevalence and to test the linear range of transcript detection, which spanned five orders of magnitude.

In yeast, NextGen expression analysis has demonstrated a larger, more complex transcriptome than had been expected (13). An estimated 74.5% of the non-repetitive yeast genome was shown to be transcribed, as were many overlapping genes, alternative initiation codons and upstream open reading frames of yeast genes were demonstrated using short reads to generate a high-resolution map of the genome. Similarly, the first high-resolution map of human genome structural variation revealed complex and large-scale structural variation in the form of insertions, deletions and inversions from a few thousand to millions of base pairs in length (14). Somatically acquired genomic rearrangements have been implicated in cancer development. Paired-end read pairs that did not align correctly with respect to each other on the reference human genome, were used to characterize 306 germline structural variants and 103 somatic rearrangements to the base-pair level of resolution (15). The results demonstrate the feasibility of using NextGen sequencing for the systematic, genome-wide characterization of rearrangements in human cancer genomes. At the level of large-scale genomic variation, copy number variations (CNV) remain difficult to measure although CNVs of 100 kilobases and greater contribute substantially to genomic variation between normal humans (16, 17). Microarray-based approaches for detecting CNVs depend on microarray signal intensity differences to predict regions of variation and cannot detect inversions. Before the advent of NextGen CNV technique, only a small fraction of copy-number variant (CNV) base pairs had been determined at the sequence level (18). NextGen CNV mapping allows the discovery cancer-causing genes in genomic regions that show recurrent copy-number alterations (gains and losses) in tumor genomes (19).

Other applications include discovering non-coding RNAs (e.g., miRNAs), sequencing the nuclear genomes of extinct species and metagenomics, such as the characterization of changes in biodiversity due to climate changes. For metagenomics, the growing number of sequenced genomes enables us to interpret partial sequences obtained by direct sampling of specific environmental niches to determine which kinds of species are present. The rapid, inexpensive, and massive data production enabled by NextGen platforms has caused a recent explosion in metagenomic studies. The NIH Human Microbiome Project is one of several international efforts using metagenomic analysis to study human health and has developing the new technological and bioinformatics tools as one of its four stated goals (http://www.genome.gov/25521743). New bioinformatics tools for assembling metagenomics data (http://nihroadmap.nih.gov/hmp/fundedresearch.asp) are required to assembling and finding genes and genomic variation in heterogeneous metagenomic datasets, a task that cannot be handled by currently available software.

Alongside the profound impact of NextGen applications in basic research, high throughput sequencing is now being adopted by clinical diagnostics laboratories for applications requiring deep sequence coverage and high-sensitivity such as rare HIV drug resistant variant detection (20). As the focus in human genetics has shifted to complex, multi-gene diseases, there is an increasing need for comprehensive diagnostic evaluations of multiple genes, enhanced by sequence enrichment/capture methods. NextGen analysis of placental mRNA - counting the numberof reads that map to each chromosome – has been used to confirm trisomy 21pregnancies, with additional supporting evidence obtained fortrisomy 18 and 13 pregnancies (REF). Other novel applications include the sequencing of ancient DNA samples and large-scale metagenomic analysis of environmentally derived samples.

**NextGen Bioinformatics**

The anticipated growth of NextGen sequencing among clinical diagnostics labs must be accompanied by streamlined sample preparation methods and improved robustness through characterization of accuracy in validation studies [6]. Biomedical research labs also require methods for mitigating systemic bias in next generation data analysis. Particularly in the early phases of the development of NextGen technology while many competing algorithms vie for supremacy, scientific publications will require comparisons of results using several different sequence analysis algorithms. At present there are insufficient studies comparing the efficacy and applicability of the various tools. An objective, performance-based comparison of NextGen bioinformatics tools is an important step towards lowering the bioinformatics hurdle and allowing smaller labs to do big science. Bioinformatics tools are available for reference alignment, de novo assembly, variant-discovery and alignment viewing. Among the reference aligners are Eland (GAPipeline v0.30, Illumina), Mira (21), Genomics Workbench (CLC Bio), Seqman NGen (DNAStar), NextGene (Soft Genetics), MAQ (22) (23) and Shrimp http://compbio.cs.toronto.edu/shrimp). De novo assemblers include Edina (24), EULER-SR (25), SHARCGS (26), SSAKE (27),Velvet (28), and SOAPdenovo (http://soap.genomics.org.cn). Some NextGen statistical data-analysis tools are also available, such as JMP Genomics (<http://www.jmp.com/software/genomics>).

Despite the growing number of software packages available for NextGen data, obtaining an accurately assembled sequence contig is a still a very challenging problem. The currently available assembly/alignment programs vary widely in terms of data volume capacity (e.g., bacterial versus human data sets), number of reads aligned/assembled, error rates and bias, all of which may lead to suboptimal assemblies. A performance-based comparison of these tools is an important precondition for mitigating systemic bias in next generation data analysis. One anticipated outcome of this will be hastening the transition to a mature technology, with fewer bioinformatics applications used for a wide number of applications. In some cases, extensive comparisons may be needed to determine that certain bioinformatics approaches are suitable for a particular task however to determine this investigators can benefit from having all the applications at their fingertips.

Other bioinformatics challenges include the development of quality value standards similar to that of phred in Sanger sequencing (7) and the development of NextGen sequence viewers. As the bioinformatics challenge shifts from mastering the tools to gaining biological insights using huge amounts of data, visualization and interpretation of short read data become paramount. The three commercial software packages by DNAStar, SoftGeneticsand CLC Bio contain data viewers that allow the user to see read alignments, coveragedepth, genome annotations, and variant analysis. However, they as yet lack the capability for viewing data sets as large as a whole human chromosome and show performance losses even on sub-chromosome data sets. The three major genome viewers – UCSC Genome Browser (REF), Ensembl genome browser (REF) and GBrowse (REF) – are based on the thin-client model where user actions result in a reloaded image file sent from the server. Java-based applications such as Apollo (REF) lack a concerted approach to data sharing although newer applications IGV (REF) do allow for limited filtering of the displayed features (ADD MORE LATER).

At the other end of the spectrum, the few studies on common bioinformatics tasks (Stephens, 2001) and usability of bioinformatics tools (Colchini, 2007) identified a lack of workflow-based tools for bioscience researchers. The majority of bioinformatics workflow tools only partially meet the need to be able to routinely design and run complex and robust workflows using multiple applications and data sets.

Bare, 2007 Firegoose

***Comparison with workflow software***

Firegoose, along with the Gaggle framework, shares several

features in common with workflow tools such as Taverna[

30,31]. They share the strategy of composing distinct

programs and data sources to build larger systems with

rich capabilities. To this end, both Firegoose and workflow

tools benefit greatly from the availability of programmatic

access to structured data and computational services

over common web protocols.

Workflow tools enable a user to automate a well-defined

and repeatable process, often using web services or message

queues for interprocess communication. But, before

a well-defined analysis process exists, there is a need for

exploratory analysis, which is necessarily ad-hoc. Gaggle

and the Firegoose seek to enable this kind of interactive

exploration by exploiting the flexibility of web-based

tools in combination with desktop analysis and visualization

tools. Scripting an interaction with a web site is, of

course, possible using a browser extension, but the

emphasis in the Firegoose is on automating the exchange

of data, leaving the direction of the analysis up to the user.

The difference in emphasis does not rule out using workflow

engines and Firegoose together. Invoking workflows

on a remote server from within Firegoose is one potentially

valuable example.

***Future Work***

Additional functionality could be added to the Firegoose

in a number of ways, most easily through the addition of

more handlers for biological websites. We also considered

that users might want to develop their own handler

scripts. We prototyped code for dynamically importing

custom scripts into the Firegoose. Other projects such as

Greasemonkey [8] have had success with similar capabilities.

If further developed, this feature would allow a

straightforward mechanism for users or data providers to

contribute scripts.

Supporting the RMI communications protocol requires

Java. Using Java within a Firefox extension is something of

a challenge and code from MIT's Simile project [9] was

extremely helpful in this area. An alternative under consideration

is to communicate with the Boss using an XML

based protocol over sockets eliminating the need to run a

Java virtual machine in the browser's process.

We plan to extend the Gaggle microformat to express links

to data in addition to embedding data directly in the page.

This allows large data structures to be transmitted independently

of the page while preserving the linkage

between presentation in the browser and the underlying

structured data. A standard format would decrease the

need for customized coding for each web site.

RDF (Resource Description Framework) is a data model

designed to represent meta-data for Semantic Web applications.

Incorporating support for RDF into the Firegoose

would allow the Gaggle to exchange data with the semantically

rich resources envisioned by proponents of the

Semantic Web project.

**Conclusion**

The Firegoose incorporates Mozilla Firefox into the Gaggle

environment providing coordinated access to web

applications and programmatic data sources. Performing

data integration in the browser has several advantages and

is perhaps the most interesting feature of the Firegoose.

Browsers excel at search and navigation. Using the Firegoose,

a biologist can search and navigate web resources

using familiar browser-based interfaces with the additional

capability of easily moving data from one webbased

resource to another as well as between the web and

the desktop. Interactively integrating specific information

as needed replaces the cumbersome process of maintaining

local copies of large databases and manually coercing

data from diverse sources into a compatible format. Using

the Gaggle data types as intermediaries lowers the barrier

between web resources and desktop tools, allowing the

scientist to creatively combine and re-use data in ways that

go beyond those provided by the curators of individual

data sources.

The Firegoose positions the Gaggle to take advantage of

increasing use of web protocols to transmit structured

data. The Firegoose provides a framework in which new

web resources can be integrated into the Gaggle in a

straightforward and easily implemented manner, accommodating

a variety of protocols. In supporting a number

of protocols, we hope to encourage data providers to

make available structured data in the format of their

choice and to provide the necessary information to link

web interfaces with the underlying data allowing browsing

and programmatic access to become seamlessly integrated.

If the web is becoming a channel for structured data,

applications that share data between diverse web

resources and software tools will be of increasing importance.

The Firegoose aims to fill this role for the systems

biology domain.

**Availability and requirements**

Source code for the Firegoose, along with that of the other

components of the Gaggle, is available at the Gaggle website

[32]. Also available are instructions for installing and

uninstalling the Firegoose toolbar [33] and documentation

[34]. Most of the desktop components of the Gaggle

are deployed as Java webstarts, which can be launched by

clicking a link in the browser.

The toolbar is compatible with versions 1.5.x and 2.0.x of

Mozilla Firefox. We anticipate maintaining compatibility

with Firefox 3.x when released.

Java version 5 [35] or higher runtime environment is

required and the Java browser plug in for Firefox must be

installed. Extra attention is often required to install the

Java browser plug-in on Linux. Specific instructions for

most distributions are available on the web.

**The source code is distributed under the GNU Lesser General**

**Public License, the text of which is available at: http:/**

**/www.gnu.org/copyleft/lesser.html.**

Firegoose download

<http://gaggle.systemsbiology.net/docs/geese/firegoose/>

Creating a bioinformatics nation

<http://www.nature.com/nature/journal/v417/n6885/full/417119a.html>

Distiction between analysis workflows and production workflows

Tilson 2007 Workflows ppt

Taverna, Complex workflows

* Service Technologies
  + BioMoby
  + Seqhound
  + Soaplab
  + WSDL
  + Custom
  + Etc.
* Taverna can support all of these ( and more)
  + **Develop useful Data-flow workflows**

Shannon 2006 Gaggle

Guided by the classic

software engineering strategy of *separation of concerns* and a policy of *semantic flexibility*, it integrates

existing popular programs and web resources into a user-friendly, easily-extended environment

Tambis Manchester

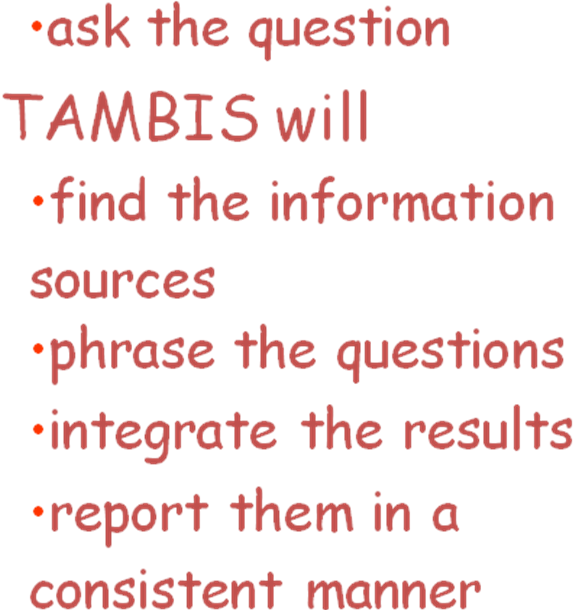
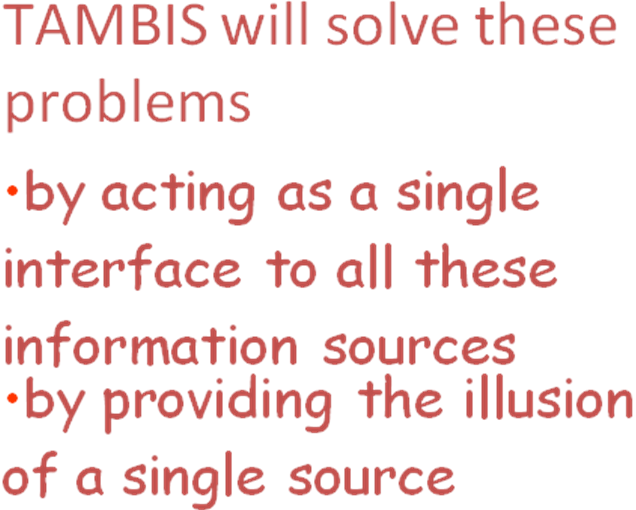
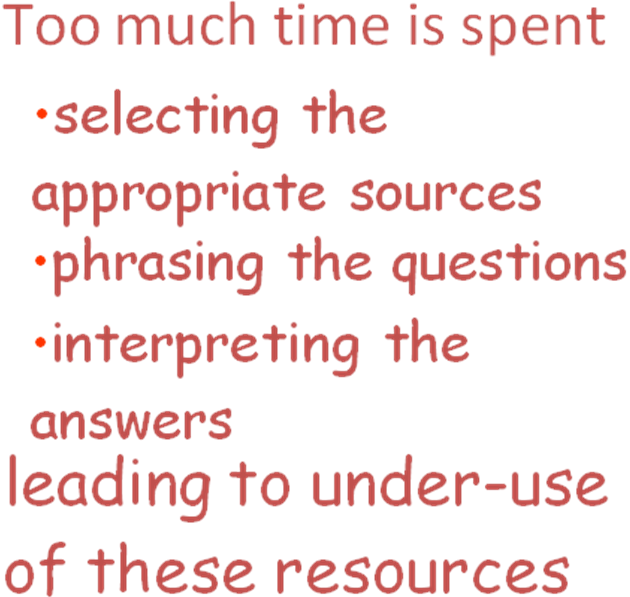
There are three main types of information source: databases, online services, files (Tambis ppt).

Researchers in biological sciences often gather and compare sequencing and other heterogeneous information

This process takes significant manual effort ...

... and is a constant barrier to progress

There are over 200 biological information sources world-wide



# Workflows in bioinformatics: meta-analysis and prototype implementation of a workflow generator

**Alexander Garcia Castro**1,2 , **Samuel Thoraval**2,3 , **Leyla J Garcia**4 and **Mark A Ragan**1,2



BMC Bioinformatics 2005, **6:**87doi:10.1186/1471-2105-6-87

### Abstract

#### Background

Computational methods for problem solving need to interleave information access and algorithm execution in a problem-specific workflow. The structures of these workflows are defined by a scaffold of syntactic, semantic and algebraic objects capable of representing them. Despite the proliferation of GUIs (Graphic User Interfaces) in bioinformatics, only some of them provide workflow capabilities; surprisingly, no meta-analysis of workflow operators and components in bioinformatics has been reported.

#### Results

We present a set of syntactic components and algebraic operators capable of representing analytical workflows in bioinformatics. Iteration, recursion, the use of conditional statements, and management of suspend/resume tasks have traditionally been implemented on an ad hoc basis and hard-coded; by having these operators properly defined it is possible to use and parameterize them as generic re-usable components. To illustrate how these operations can be orchestrated, we present GPIPE, a prototype graphic pipeline generator for PISE that allows the definition of a pipeline, parameterization of its component methods, and storage of metadata in XML formats. This implementation goes beyond the macro capacities currently in PISE. As the entire analysis protocol is defined in XML, a complete bioinformatic experiment (linked sets of methods, parameters and results) can be reproduced or shared among users. Availability: <http://if-web1.imb.uq.edu.au/Pise/5.a/gpipe.html> [webcite](http://www.webcitation.org/query.php?url=http://if-web1.imb.uq.edu.au/Pise/5.a/gpipe.html&refdoi=10.1186/1471-2105-6-87) (interactive), <ftp://ftp.pasteur.fr/pub/GenSoft/unix/misc/Pise/> [webcite](http://www.webcitation.org/query.php?url=ftp://ftp.pasteur.fr/pub/GenSoft/unix/misc/Pise/&refdoi=10.1186/1471-2105-6-87) (download).

#### Conclusion

**From our meta-analysis we have identified syntactic structures and algebraic operators common to many workflows in bioinformatics. The workflow components and algebraic operators can be assimilated into re-usable software components. GPIPE, a prototype implementation of this framework, provides a GUI builder to facilitate the generation of workflows and integration of heterogeneous analytical tools.**

**An ontology-based framework for bioinformatics**

**workflows**

Luciano A. Digiampietri

**Abstract:**

The proliferation of bioinformatics activities brings new challenges

– how to understand and organise these resources, how to exchange and

reuse successful experimental procedures, and to provide interoperability

among data and tools. This paper describes an effort toward these directions.

It is based on combining research on ontology management, AI and

scientific workflows to design, reuse and annotate bioinformatics experiments.

The resulting framework supports automatic or interactive composition of tasks

based on AI planning techniques and takes advantage of ontologies to support

the specification and annotation of bioinformatics workflows. We validate our

proposal with a prototype running on real data.

…

We use domain ontologies as the basis for attacking these problems. The first

question – provenance of data and software tools – directly affects the acceptance of the

results of experiments. The quality of bioinformatics experiments depends on properly

identifying data origins and the processes that produced these data (Buttler et al., 2002).

At most times, provenance is indicated by laborious manual annotations, which often

vary across laboratories.

The second issue concerns tool/task composition while constructing the workflows

(Cavalcanti et al., 2005; Yu and Buyya, 2005; Medeiros et al., 2005). We highlight three

kinds of composition: manual (supervised), iterative (using top-down design practices)

and automatic. In a scenario where several software tools are being made available on the

web, the composition problem has become more important. To help this issue, many tools

invoked by such workflows are now encapsulated into web services

Aqwa gamma may make use of existing syntactic structures and algebraic operators common to bioinformatics workflows (Garcia et al., 2005) to achieve fully customizable workflows, with forks determined by conditional statements or otherwise and treelike workflows composed of multiple workflows linked together with logical commands. Alternately, since these workflow command constructs are based on XML, a different approach may be more suitable for JSON, which is at the core of the Aqwa design. A thorough evaluation of the efficacy of existing methods and applicability to the Aqwa design will be carried out prior to Phase 3 (Aqwa gamma) development.

In summary, high throughput sequencing requires high throughput analysis. Although the falling cost of NGS sequencing has allowed a widening pool of users to access the technology, the costs of downstream bioinformatics analysis resources is often underestimated. Constraints on grant revenues and a shortage of qualified personnel mean that bioinformatics resources are scarce in comparison to the increasing rate of generation of NGS sequence data. Even provided the necessary funding and personnel, building bioinformatics capacity is a time-consuming process which can extend many months after sequence data becomes available.

**Preliminary studies**

**1. Evaluation of next gen assemblers**

We compared several commonly used short read assembly tools and propose a method for reducing these errors by combining different assemblies for the final result.

**METHOD**

Human mtDNA and whole-genome mRNA short reads produced using the Illumina/Solexa Genome Analyzer I platform were used, as well as E. coli, Herpes simplex and bacteriophage PhiX. Seven commercial and open-source short read assemblers were first assessed for assembly capacity in terms of the maximum number of reads that can be effectively assembled using relatively high-end computer hardware. We investigated the performance of Eland (GAPipeline v0.30, Illumina), Velvet v0.7.16 (28), Mira v2.9.25 (21), Genomics Workbench (CLC Bio) v1.2, Seqman NGen (DNAStar) 1.1, NextGene (Soft Genetics) 1.0 and MAQ v 0.6.8 (22) (23). Assemblies produced by the different programs were compared and a consensus determined based on read identity and divergence from the relevant reference sequence. The overall combination of assemblies was viewed for quality control purposes using a sequence viewer that we developed to handle the huge data depth and breadth of sample types.

**RESULTS**Our results indicate that there are significant differences in the capabilities of the different reference and *de novo* short read assembly tools as shown in Figures 1, 2 and 3. The resulting assemblies showed significant differences in read matching against the reference sequence in particular locations. For human data, Seqman NGen, Genomics Workbench and NextGen showed better performance in terms of the number of reads assembled however this may result in less accurate contigs. A combination of different assemblies can provide more reliable estimates of genetic aberrations by flagging dubious assembly regions that are not represented in a majority of the different assemblies. Conversely, regions that are matched identically by a majority of the different algorithms can be accorded greater confidence with regard to their predicted SNPs, indels and breakpoints. Based on this research, we are currently developing a new visualization and analysis tool to meet the needs of next generation sequencing data analysis.

**CONCLUSION**

Effective use of these technologies depends on the correct interpretation of differing assembly results based on a consensus of reads incorporated by the various assembly algorithms. Manual verification of the assembly combination using a Next Generation sequence viewer improved confidence in the resulting assembly and aided comprehension of the strengths and weaknesses of this approach.

**2. Aqwa: Automated Query and Workflow Agent**

**Software Design Document**

**Introduction**

1. Project-based structure
2. Data sharing among groups
3. Report extraction and sharing
4. External data (API-based query, custom queries)
5. File format conversion
6. ID conversion - DAVID (29)
7. Transcriptome
8. SNP verification
9. Genetic network analysis (GeneSet Analyzer - Nick)
10. Integration with Pathway Analysis software, e.g., Cytoscape, GeneGo/MetaCore and Systems Biology modeling software

Data design (API, SQL tables)

Interface design

Risks

At the root of most UML fevers is a lack of practical experience in those individuals responsible for selecting and applying the technologies and processes underlying a program’s software-development efforts. This lack of experience translates into both unrealistic expectation and misapplication of technology, often aggravated by nonexistent or bad software-development processes, a perfect breeding ground for UML fever. If a software organization’s battle against UML fever is to be successful, it is absolutely critical that people with practical experience are in place driving the selection of technologies, as well as developing the processes for their associated usage.

**Research design and methods**

*How key variables will be defined and measured, and the general approach to analysis of the data.*

**Software design strategy**

1. Human-centered development process
2. Classification of bioinformatics tasks
3. General query and workflow requirements
4. Usability analysis (e.g., User tests)(30)

**Feasibility (***Discussion of the strengths and weaknesses of the proposed study)*

Goes here…

**Aqwa roadmap (***Realistic time line***)**

Goes here…

**Hardware requirements** (Joel)

8-node (8-core each) cluster

HPC cluster time

Production server

Development server

**PHS 398 Research Plan structure and format**

**(part of SF 424 (R&R) Application for Federal Assistance)**

*1. Introduction to Application* ***XXX NOT REQUIRED*** *(for RESUBMISSION or REVISION only)*

2. Specific Aims

3. Background and Significance

4. Preliminary Studies / Progress Report

5. Research Design and Methods

Notes on Required Format ( <http://grants.nih.gov/grants/funding/424/SF424_RR_Guide_General_Adobe_VerA.doc>)

Text attachments should be generated using word processing software and then converted to PDF using PDF generating software. Additional tips for creating PDF files can be found at <http://era.nih.gov/ElectronicReceipt/pdf_guidelines.htm>.

When attaching a PDF document to the actual forms, please note you are attaching an actual document, not just pointing to the location of an externally stored document. Therefore, if you revise the document after it has been attached, you **must** delete the previous attachment and then reattach the revised document to the application form. Use the “**View Attachment**” button to determine if the correct version has been attached.

**Font:** Use an Arial, Helvetica, Palatino Linotype, or Georgia typeface

**Color:** Black

**Size:** 11 points or larger. (A Symbol font may be used to insert Greek letters or special characters; the font size requirement still applies.)

**Type density:** including characters and spaces, must be no more than 15 characters per inch.Type may be no more than six lines per inch.

**Page Margins:** Use standard paper size (8 ½" x 11). Use at least one-half inch margins (top, bottom, left, and right) for all pages. No information should appear in the margins, including the PI’s name and page numbers.

**Header/footer:** Do not include any information in a header or footer of the attachments. Page numbers for the footer will be system-generated in the complete application, with all pages sequentially numbered.

**Figures, Graphs, Diagrams, Charts, Tables, Figure Legends, and Footnotes**

You may use a smaller type size but it must be in a black font color, readily legible, and follow the font typeface requirement. Color can be used in figures; however, all text must be in a black font color, clear and legible.

**Acronyms/Abbreviations:** If terms are not universally known, spell out the term the first time it is used and note the appropriate abbreviation in parentheses. The abbreviation may be used thereafter.

**Separate Attachments**

Separate attachments have been designed for the Research Plan sections to maximize automatic validations conducted by the eRA system. When the application is received by the agency, all of the Research Plan sections will be concatenated in the appropriate order so that reviewers and agency staff will see a single cohesive Research Plan.

While each section of the Research Plan needs to eventually be uploaded separately, applicants are encouraged to construct the Research Plan as a single document, separating sections into distinct PDF attachments just before uploading the files. In this way the applicant can better monitor formatting requirements such as page limits. When validating for page limits, the eRA Commons will not count the white space created by breaking the text into separate files for uploading.

**Page Limits**

Although many of the sections of this application are separate text (PDF) attachments, page limitations referenced in these instructions and/or funding opportunity announcement must still be followed. Agency validations will include checks for page limits. Some accommodation will be made for sections that when combined must fit within a specified limitation. Note that while these computer validations will help minimize incomplete and/or non-compliant applications, they do not replace the validations conducted by NIH staff. Applications found not to comply with the requirements may lead to rejection of the application during agency validation or delay in the review process.

All applications and proposals for NIH and other PHS agency funding must be self-contained within specified page limitations. Unless otherwise specified in an NIH solicitation, Internet website addresses (URLs) may not be used to provide information necessary to the review because reviewers are under no obligation to view the Internet sites. Moreover, reviewers are cautioned that they should not directly access an Internet site as it could compromise their anonymity.

1. NHGRI. NHGRI Seeks DNA Sequencing Technologies Fit for Routine Laboratory and Medical Use. 2008 [updated 2008; cited]; Available from: <http://www.genome.gov/27527585>.

2. Mardis ER. Next-Generation DNA Sequencing Methods. Annual Review of Genomics and Human Genetics. 2008;9(1):387-402.

3. Schuster SC. Next-generation sequencing transforms today's biology. Nat Meth. 2008;5(1):16-8.

4. Bennett S. Solexa Ltd. Pharmacogenomics. 2004;5(4):433-8.

5. Margulies M, Egholm M, Altman WE, Attiya S, Bader JS, Bemben LA, et al. Genome sequencing in microfabricated high-density picolitre reactors. Nature. 2005;437(7057):376-80.

6. Shendure J, Porreca GJ, Reppas NB, Lin X, McCutcheon JP, Rosenbaum AM, et al. Accurate Multiplex Polony Sequencing of an Evolved Bacterial Genome. Science. 2005;309(5741):1728-32.

7. Shendure J, Ji H. Next-generation DNA sequencing. Nat Biotech. 2008;26(10):1135-45.

8. Bashiardes S, Veile R, Helms C, Mardis ER, Bowcock AM, Lovett M. Direct genomic selection. Nat Methods. 2005;2(1):63-9.

9. Okou DT, Steinberg KM, Middle C, Cutler DJ, Albert TJ, Zwick ME. Microarray-based genomic selection for high-throughput resequencing. Nat Methods. 2007;4(11):907-9.

10. Fredriksson S, Baner J, Dahl F, Chu A, Ji H, Welch K, et al. Multiplex amplification of all coding sequences within 10 cancer genes by Gene-Collector. Nucleic Acids Res. 2007;35(7):e47. PMCID: 1874629.

11. 't Hoen PAC, Ariyurek Y, Thygesen HH, Vreugdenhil E, Vossen RHAM, de Menezes RX, et al. Deep sequencing-based expression analysis shows major advances in robustness, resolution and inter-lab portability over five microarray platforms. Nucl Acids Res. 2008;36(21):e141-.

12. Mortazavi A, Williams BA, McCue K, Schaeffer L, Wold B. Mapping and quantifying mammalian transcriptomes by RNA-Seq. Nat Methods. 2008;5(7):621-8.

13. Nagalakshmi U, Wang Z, Waern K, Shou C, Raha D, Gerstein M, et al. The Transcriptional Landscape of the Yeast Genome Defined by RNA Sequencing. Science. 2008;320(5881):1344-9.

14. Kidd JM, Cooper GM, Donahue WF, Hayden HS, Sampas N, Graves T, et al. Mapping and sequencing of structural variation from eight human genomes. Nature. 2008;453(7191):56-64. PMCID: 2424287.

15. Campbell PJ, Stephens PJ, Pleasance ED, O'Meara S, Li H, Santarius T, et al. Identification of somatically acquired rearrangements in cancer using genome-wide massively parallel paired-end sequencing. Nat Genet. 2008;40(6):722-9.

16. Sebat J, Lakshmi B, Troge J, Alexander J, Young J, Lundin P, et al. Large-scale copy number polymorphism in the human genome. Science. 2004;305(5683):525-8.

17. Iafrate AJ, Feuk L, Rivera MN, Listewnik ML, Donahoe PK, Qi Y, et al. Detection of large-scale variation in the human genome. Nat Genet. 2004;36(9):949-51.

18. Cooper GM, Nickerson DA, Eichler EE. Mutational and selective effects on copy-number variants in the human genome. Nat Genet. 2007;39(7 Suppl):S22-9.

19. Chiang DY, Getz G, Jaffe DB, O'Kelly MJ, Zhao X, Carter SL, et al. High-resolution mapping of copy-number alterations with massively parallel sequencing. Nat Methods. 2009;6(1):99-103. PMCID: 2630795.

20. Wang C, Mitsuya Y, Gharizadeh B, Ronaghi M, Shafer RW. Characterization of mutation spectra with ultra-deep pyrosequencing: application to HIV-1 drug resistance. Genome Res. 2007;17(8):1195-201. PMCID: 1933516.

21. Chevreux B, Pfisterer T, Drescher B, Driesel AJ, Müller WEG, Wetter T, et al. Using the miraEST Assembler for Reliable and Automated mRNA Transcript Assembly and SNP Detection in Sequenced ESTs. Genome Res. 2004;14:1147-59.

22. Li H, Durbin R. MAQ: Mapping and Assembly with Qualities. 2007 [updated 2007; cited 2008 November 11th]; Available from: <http://maq.sourceforge.net/>.

23. Li H, Ruan J, Durbin R. Mapping short DNA sequencing reads and calling variants using mapping quality scores. Genome Res. 2008;18(11):1851-8. PMCID: 2577856.

24. Hernandez D, Francois P, Farinelli L, Osteras M, Schrenzel J. De novo bacterial genome sequencing: millions of very short reads assembled on a desktop computer. Genome Res. 2008;18(5):802-9. PMCID: 2336802.

25. Chaisson MJ, Pevzner PA. Short read fragment assembly of bacterial genomes. Genome Res. 2008;18(2):324-30. PMCID: 2203630.

26. Dohm JC, Lottaz C, Borodina T, Himmelbauer H. SHARCGS, a fast and highly accurate short-read assembly algorithm for de novo genomic sequencing. Genome Res. 2007;17(11):1697-706. PMCID: 2045152.

27. Warren RL, Sutton GG, Jones SJ, Holt RA. Assembling millions of short DNA sequences using SSAKE. Bioinformatics. 2007;23(4):500-1.

28. Zerbino DR, Birney E. Velvet: Algorithms for de novo short read assembly using de Bruijn graphs. Genome Res. 2008;18:821-9.

29. Huang da W, Sherman BT, Stephens R, Baseler MW, Lane HC, Lempicki RA. DAVID gene ID conversion tool. Bioinformation. 2008;2(10):428-30. PMCID: 2561161.

30. Bolchini D, Finkelstein A, Perrone V, Nagl S. Better bioinformatics through usability analysis. Bioinformatics. 2009;25(3):406-12.