**2. Specific Aims (one page maximum; separate PDF attachment)**

NextGen sequencing technologies are fast approaching the ‘$1,000 genome’ target (1): a $5,000 genome will be available in May 2009 by Comparative Genomics while other NextGen industry players are rapidly 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). Distributed processing on high performance computing clusters is necessary in order to cope with the large data volumes and reduce the cost of processing. Furthermore, costly investments in skilled personnel are also required to develop, evaluate and run bioinformatics algorithms, and to integrate diverse biological data sources. Most biomedical research and diagnostics labs are unable to provide even the minimum of these hardware and personnel requirements. With regard to software, workflow tools are essential to allow non-technical staff to automate and run well-defined but complex analysis processes. These tools must also be flexible enough to support exploratory analysis through interaction with the data using a wide range of different software applications and data processing steps. Ideally, workflow tools should also be web-enabled for flexibility of access by users and integration with other bioinformatics web resources. They should also provide visualization functionality capable of handling large volumes of NextGen data and integrating heterogeneous external genome feature data sets. Given the budget considerations mentioned above, the ideal workflow tool should also be open source and freely available to the academic community. At present, there are no tools available that span the whole process from NextGen sequence generation to analysis and visualization.

To help address these opportunities, we propose the rapid deployment of a software system and analysis tools for managing NextGen sequencing projects, from short read generation to bioinformatics analysis to data visualization. The system will meet the following challenges: 1) facilitating the analysis of large-scale sequencing studies, 2) enabling transcriptome and genetic network analyses, and 3) determination of the relationship of sequence variation and phenotypes to disease. These challenges will be addressed through the following specific aims:

**Specific Aim 1: Implement an optimized assembly workflow**

We propose carrying out an objective evaluation of current NextGen assemblers/aligners. Based on this assessment, we will provide an optimized workflow for each of the three main NextGen sequence platforms (Illumina/Solexa, Roche/454 and ABI/SOLiD) to generate assemblies and their associated quality control information. These workflows will be customizable by the user to suit their particular desired quality metrics or tradeoffs.

**Specific Aim 2: Implement variation annotation, expression analysis and network analysis workflows**

We propose developing a variation annotation pipeline with defined quality control/assurance algorithms for verifying and annotating SNPs (single nucleotide polymorphisms), large-scale structural variation and copy number variation. The pipeline will be integrated with current expression analysis packages and will develop new expression analysis algorithms. As part of the reporting and visualization of results, data filters will be designed based on user requirements to extract result subsets and provide genome-level views of the results integrated with external genomic features. Results will also be exportable to downstream analysis applications (Cytoscape, Genespring, R, etc.).

**Specific Aim 3: Develop a NextGen workflow and visualization tool**Based on the requirements in Aims 1 and 2, we propose the implementation of a novel kind of tool providing end-to-end integrated NextGen data analysis workflows and real-time genomic visualization of huge data sets. The tool will provide pre-optimized workflows for assembly/alignment, sequence and structural variation, expression analysis and network analysis and will also allow users to create their own customized workflows. The software development process will implement a user-centric approach including extensive pre- and post-release user testing at each project milestone to ensure improved usability compared to currently available tools.

**5. Research Design and Methods (12 pages maximum; separate PDF attachment)**

**Challenge Area:** *06: Enabling Technologies***Challenge Topic:** *06-HG-101\*   
New computational and statistical methods for the analysis of large data sets from next-generation sequencing technologies.*

**The Challenge and Potential Impact:** *What is the research opportunity, scientific knowledge gap or technology that will be addressed? How broad is the potential impact in science and/or health? Which community (ies) will be affected? What is (are) the size(s) of the community(ies)? Will the potential impact be major?*

This section describes the particular challenges of NextGen technologies, applications and bioinformatics in more detail and discuss the anticipated impact of solutions provided by this study.

**Specific Aim 1: Implement an optimized assembly workflow**

The impact of achieving Aim 1 will be to accelerate the development of NextGen sequencing assembly tools by providing an improved methodology for sequence assembly/alignment and the tools to do so on a routine basis. **Objective performance comparisons of assemblers/aligners involving preset criteria and using a wide range of data sets will provide a basis for comparisons between assemblers.** **Optimized pipelines based on these evaluations will allow users to improve their work efficiency and the quality of their results.** At the other end of the spectrum, customizable pipelines will help meet the bioinformatics challenges faced by researchers at the cutting edge life science exploration. Dynamic pipeline configuration coupled with high performance computing will enable researchers and other end users to rapidly develop and adapt different approaches to solving particular problems.

**NextGen Technologies**

In order to appreciate the particular problems and challenges of NextGen sequence assembly, we must first grasp the particular strengths and weaknesses of the different sequencing 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. 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 1) and smaller labs and molecular diagnostics facilities participating in growing numbers. A common refrain among adaptors of this technology is that the downstream bioinformatics analysis are often poorly understood and underestimated. **This study aims to reduce bioinformatics costs for research and diagnostics labs by providing a free, customizable tool for accomplishing all common tasks in NextGen sequence analysis.** Extensive training for users will not be required as only a basic familiarity with web page navigation and drag and drop user interfaces is assumed.

Alongside the rollout of NextGen sequencing platforms, 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 challenges and greater data infrastructure demands. Our proposed workflow and visualization tool is designed to incorporate additional applications in a “plug ‘n play” and scalable manner in order to flexibly adapt to changing computational needs.**



Figure 2. NextGen Sequencing Technology Roadmap

**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 researchers to answer more penetrating questions more convincingly and in less time. 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 (11), Genomics Workbench (CLC Bio), Seqman NGen (DNAStar), NextGene (Soft Genetics), MAQ (12) (13) and Shrimp http://compbio.cs.toronto.edu/shrimp). De novo assemblers include Edina (14), EULER-SR (15), SHARCGS (16), SSAKE (17),Velvet (18), 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 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. Little is known about the comparative performance of the available tools because the scarce available performance statistics are based mainly on different non-human data sets results (e.g., phage, bacteria, yeast). So comparison between assemblers is difficult even before considering the particular performance with human data.

**A performance-based comparison of the different assembly/alignment 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 range of applications. In some cases, extensive comparisons may be needed to determine that certain bioinformatics approaches are suitable for a particular task. **This study proposes to facilitate multi-tool comparisons by providing a workflow system to enable high-throughput assembly with multiple algorithms in parallel.**

**Specific Aim 2: Implement variation annotation, expression analysis and network analysis workflows**

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 (19). NextGen sequencing identified antisense transcription, which microarrays cannot detect, in 51% of all genes. In yeast, NextGen expression analysis has demonstrated a larger, more complex transcriptome than had been expected (20). 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. **Transcriptomes for mouse brain, liver and skeletal muscle were mapped by NextGen deep sequencing (21), 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.**

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 (22). As the focus in human genetics has shifted to complex, multi-gene diseases, there is an increasing need for comprehensive diagnostic evaluations of SNPs (single nucleotide polymorphisms) and other genomic variation in multiple genes. 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 (23). 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 (24). 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 (25, 26). 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 (27). NextGen CNV mapping allows the discovery of cancer-causing genes in genomic regions that show recurrent copy-number alterations (gains and losses) in tumor genomes (28).

**Specific Aim 3: Development of a NextGen workflow and visualization tool**

As bioscience becomes increasingly a quantitative analysis activity, workflow tools enable users to accomplish two main tasks: 1) automating well-defined, repetitive processes and 2) exploring data with ad-hoc analyses (29). The few studies on common bioinformatics tasks (Stephens, 2001) and usability of bioinformatics tools (Colchini, 2007) identified an unmet need among bioscience researchers for workflow-based tools. There are over 200 major Internet biological data sources however these sites are mostly simple GUIs (Graphical User Interfaces) with limited data compatibility between them despite the fact that researchers often need to combine the outputs of multiple sites to generate bioinformatics analyses. **These resources may be underexploited if users feel too much time is used navigating the sites, selecting among appropriate sources, downloading and uploading files. Workflow tools can solve this problem by generating large and complex systems from collections of programs, data sources and even structured data services. However, the majority of bioinformatics workflow tools only partially realize the potential of the available data and application resources (Figure 2).** With notable exceptions (30), there has been limited progress in connecting different sites with the client as the intermediary. Another approach is to use a central site as a service directory lookup such as BioMOBY (31) and TAVERNA (32) with the limitation that service discovery relies upon the accurate and uniform description of biological data types and relations between them, for which there is no commonly-accepted ontology (33-35) or language, despite some developments (36).

Data visualization and interpretation become paramount as the bioinformatics challenge shifts from mastering the basic tools to gaining biological insights from huge amounts of data. Three commercial software packages by DNAStar, SoftGeneticsand CLC Bio provide data viewers that allow the user to see read alignments, coveragedepth, genome annotations, and variant analysis. However, they lack the capability for viewing data sets as large as a whole human chromosome and show poor performance even on sub-chromosome data sets. The three major publicly available genome viewers – UCSC Genome Browser (37, 38), Ensembl genome browser (39) and GBrowse (40, 41) – are based on the traditional client-server model where the user’s requested data is reloaded as an image file delivered from the server. Java-based applications such as Apollo (42, 43) are more interactive but lack a concerted approach to data sharing. Most do not allow the user to filter the displayed data set based on biological criteria, although some newer applications such as IGV (http://www.broad.mit.edu/igv/) allow for limited filtering of the displayed features. **Figure 2 lists the capabilities that a fully functional workflow tool and genome viewer must possess and shows the gaps in functionality of the currently available tools. We propose to provide all of this functionality in an integrated workflow tool and genome viewer.**



Figure 2. An overview of workflow and viewer functionalities of currently available workflow tools and genome viewers.

**The Approach:** *How will you attempt to explore or solve the stated research problem? How will your rationale and/or approach overcome existing challenges or barriers in the field? If you propose to improve existing technologies or to develop new technologies, which needs are being addressed and what is unconventional and exceptionally innovative about your approach? Provide enough information for reviewers to determine what you are proposing to do, but do not include a detailed experimental plan.*

We propose to address the abovementioned bioinformatics challenges, namely the development of 1) improved NextGen sequence assembly workflows, 2) optimized variation, transcriptome and genetic interaction pathways, 3) a NextGen workflow and visualization tool with the following approaches:

**Specific Aim 1: Implement an optimized assembly workflow**

We propose carrying out an objective evaluation of current NextGen assemblers/aligners using artificial data sets based on human biological samples in which each read’s position is known *a priori* in order to accurately compare results between different algorithms. Performance criteria will be established before testing based on the particular difficulties of assembling short reads derived from human genomic material. One or more optimized assembly workflows maximizing the performance criteria will be created as push-button tools to generate assemblies and associated quality control information. These workflows will also be customizable by the biologist/researcher to suit particular desired quality metrics or to meet any necessary tradeoffs between different quality metrics.

As the first stage of this specific aim, we investigated the performance of the following short read assembly tools: Eland (GAPipeline v0.30, Illumina), Velvet v0.7.16 (18), Mira v2.9.25 (11), Genomics Workbench (CLC Bio) v1.2, Seqman NGen (DNAStar) 1.1, NextGene (Soft Genetics) 1.0 and MAQ v 0.6.8 (12) (13). The input data sets were: human mtDNA, human whole-genome mRNA, E. coli DNA, Herpes simplex and bacteriophage PhiX. The assemblers were assessed for assembly speed and capacity in terms of the maximum number of reads that can be effectively assembled using relatively high-end computer hardware. Assemblies produced by the different programs were compared and a consensus determined based on read identity and divergence from the relevant reference sequence. **There were significant differences in the sequence capacities of the different reference alignment and *de novo* short read assembly tools, and the reference aligner outputs showed significant differences in reads 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 may 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 might be accorded greater confidence with regard to their predicted SNPs, indels and breakpoints. **We will attempt to prove or disprove these hypotheses with a view to designing an optimized assembly workflow in the next step of specific aim 1 by using artificial reads generated from human sequence samples selected from the SRA (Short Read Archive** [**http://www.ncbi.nlm.nih.gov/Traces/sra/sra.cgi**](http://www.ncbi.nlm.nih.gov/Traces/sra/sra.cgi)**).** Samples from individuals with sequences from all three NextGen sequencing platforms will be selected based on coverage depth, the presence of family member samples and the availability of additional genomic analyses such as microarrays in order to validate anomalies and discriminate between differing sequence assembly results.

**Specific Aim 2: Implement variation annotation, expression analysis and network analysis workflows**

We propose developing a variation annotation pipeline with defined quality control/assurance algorithms for SNPs, large-scale structural variation and copy number variation. **The pipeline will be integrated with current expression analysis packages such as ERANGE (21) and will develop new expression analysis algorithms to improve quantization of transcript counts.** As part of the reporting and visualization of results, data filters will be designed based on user requirements to extract result subsets and provide genome-level views of the results integrated with external genomic features. Results will also be exportable to downstream analysis applications (Cytoscape, Genespring, R, etc.).

**We will use actual experimental data from multiple studies of different human diseases to validate our variation annotation pathway.** In the first stage of specific aim 2, we are developing a methodology for estimating erroneous SNP calls and predicting homozygote and heterozygote genotypes using Roche/454 reads derived from Nimblegen exome capture samples. The experiment subjects are an extended family of 8 individuals, including 4 with a heritable neurodegenerative disorder. We will validate our approach with a sample of 1,000 individuals on a smaller subset of captured genomic regions. **This analysis is being carried out in partnership with the MIHG (Miami Institute for Human Genomics).** **Further joint studies with the MIHG are anticipated involving analysis of SNPs, large-scale variation and copy number variation in several other diseases.**

**Specific Aim 3: Develop a NextGen workflow and visualization tool**Based on the requirements in Aims 1 and 2, we propose the implementation of a novel bioinformatics tool providing end-to-end integrated NextGen data analysis workflows and real-time visualization of huge genomic data sets. The tool will provide pre-optimized workflows for assembly/alignment, sequence and structural variation, expression analysis and network analysis and will also allow users to create their own customized workflows. **The software development process will implement a user-centric approach including extensive user testing at each project milestone and intervening iterations. Our emphasis on user testing and user-centric development, which differs from all previous bioinformatics workflow tools, is designed to ensure that the user interface is as intuitive as possible.**

The following list of requirements and functional criteria encompasses the functionality envisaged for the workflow and genome view tool, henceforth referred to as Aqwa (Automated Query and Workflow Agent).

**Functional Requirements**

1. Low barriers to usage
   1. Web access
   2. User-friendly, intuitive interface
   3. “Plug ‘n play” applications for rapid deployment
   4. Searchable project annotations
2. Workflows
   1. Predefined workflows (transcriptome, variation annotation, gene networks, file conversion utilities, ID conversion utilities, etc.)
   2. Customizable workflows
   3. Drag ‘n drop workflows
   4. Persistent data and workflow configurations
   5. Loops, conditional branching
3. Reports - customizable report extraction from workflow output
4. Views
   1. Customizable genomic views of report data
   2. Interactive display with rich context menu
   3. Integrated, extensible genomic features
   4. Multiple feature views – nucleotide level to aggregate high-level view
   5. Fast view update
   6. User can filter view based on data
   7. Genomic feature-level annotation by user
5. Sharing
   1. User-defined groups with customizable permissions
   2. Workflow, report and view sharing among groups
6. Input/Output and execution
   1. Import external biological data and genomic features into workflow
   2. Integration with external software (e.g., Cytoscape, R, GeneSet Analyzer)
   3. Programmatic remote access (API and Web Service)
   4. Cluster execution
   5. Grid execution
7. Maintain state (action history)
8. Data management – direct user access to input and output files

Information regarding data provenance (44) is retained in the system to identify the source of data throughout the workflow such as the owner, author application, creation and modification dates, and content type. In addition, a log is kept of all project changes and updates. The user can also annotate the project at all levels and search these annotations. Aqwa is similar to the laboratory notebook paradigm employed by the BCJ (Bioinformatics Computational Journal) workflow tools (45) but has a wider the range of functionality. **Aqwa’s functional requirements largely encompass those for a proposed ‘genome wiki’ (46) intended to facilitate cooperative genome annotation by a community of experts, reflecting Aqwa’s utility to the wider genomics community as an accurate, continually updated source of genome annotation.** Aqwa’s genome viewer functionality incorporates the AJAX-enabled JBrowse genome viewer (<http://jbrowse.org>) to provide a fast, fluid and responsive genome browser interface (Figure 3).



Figure 3. Aqwa’s Jbrowse-based allows drag ‘n drop feature selection and real-time zooming and panning.

From the user’s perspective, designing customizable workflows can be a daunting prospect due to the complexity of inputs for bioinformatics applications. Help information is displayed for each application linked to the University of Miami’s online bioinformatics information services portal (<http://bio.ccs.miami.edu/ibis>) and an automatic syntax checker ensures that application inputs are sufficient and correct. In addition, each application object contains methods that automatically derive its input arguments from the resources and outputs of preceding applications in the workflow. In the web interface, each application in a workflow is represented as a block with required input parameters and outputs (Figure 4).



Figure 4. Workflows are created by dragging applications from the left toolbar into the center pane. Users can choose to use the default automated argument settings or manually configure applications in the right pane.

For advanced users, Aqwa provides a file manager which allows direct access and drag ‘n drop manipulation of the file system of each workflow (Figure 5), rather than solely an abstract data interface such as in the BCJ. This is an optional function, which provides the benefits of fine control of workflows for power users without the burden of increased complexity for less advanced users.



Figure 5. Aqwa’s file manager allows drag ‘n drop manipulation of workflow file systems.

The various activities involved in Aqwa’s development are detailed in the following summarized software design document:

**Architectural design**

Guided by the software engineering strategy of *separation of concerns* for ease of development and maintenance, the system is essentially embodied by the Model-View-Controller design pattern, where the Model represents the data objects of the system (i.e., database tables on the server), the View is the display the user sees representing the Model and the Controller takes care of processing user-initiated events like mouse clicks and key presses to change the Model and the representation of the Model in the View. The Controller is composed of several MVC components that interact with a single Model to accomplish specific tasks such as authentication, project management and workflow. This component-based approach promotes ease of maintenance and reuse of architectural elements.

**User Interface (View)**

The central metaphor of the user interface is that of a project folder containing one or more workflows (customized or standard). Each workflow contains one or more applications and their input data. Each application has inputs, resources (which are similar to inputs but are not specified as command line arguments) and outputs (Figure 4). The user interface is a web application implemented in Javascript and based on the Dojo Javascript toolkit (<http://www.dojotoolkit.org>) which contains a rich assortment of web application components and utilities. Implementing the View as a web interface also allows for the eventual incorporation of three-dimensional graphical viewers directly inside the View for molecular dynamics simulation outputs. In addition, the user will be able to customize the View by specifying the default viewer for a particular data type. When the user elects to view a file with the specified data type, a copy of the file is downloaded to the client machine and the associated application is invoked with this input.

**Data design (Model)**The database tables residing on the server and their relationships make up the Model. These include tables for user authentication, project ownership and permissions, workflow content and execution status, and data provenance. Conversion of data into a unified data model in a standard format, such as in Pegasys (47), is not part of the functional requirements of the system, although the system will present different export and processing options for data based on its type. Individual applications are represented by JSON, similar to the resource definition XML files commonly used in systems such as GPIPE/PISE (48) and Pegasys (47). The data design is distinct from the API (Application Programming Interface), which is presented as a web service with relatively stable inputs and outputs.

**Low-level design (Controller)**The Controller, or so-called ‘business logic’ of the application, which interacts with the View and Model is composed of Javascript classes on the client and corresponding Perl modules on the server. The choice of Perl as the backend programming language was influenced by its large user community, abundance of bioinformatics tools such as BioPerl (http://www.bioperl.org), ease of accessibility for novice programmers and wide range of applications from text manipulation to system administration.

The system will use existing or novel syntactic structures and algebraic operators for describing bioinformatics workflows (Garcia et al., 2005) to achieve fully customizable workflows with forks, conditional statements and loops, and construct treelike workflows composed of multiple workflows linked together with logical commands. JSON (JavaScript Object Notation) is used as the data interchange format because, unlike XML, JSON-encoded objects need no additional parsing to define them at the object level, which allows them to be easily chained together in complex workflows. Following the evaluation of existing workflow control methods and their applicability to the Aqwa system, a core set of operators will be implemented in the release version of Aqwa.

The first two versions of Aqwa will support execution on the local server or on a cluster using the PBS (Portable Batch System) queue scheduler. An abstraction layer separates the workflows and the execution method for ease of extension in later versions to execution on a grid and eventually as web service requests. The latter two execution modes will require a more sophisticated pipe component and conditional operator due to the need to check for service availability before jobs can be dispatched for execution on a remote host. Failure management of jobs executed on a local server or cluster is accomplished by job monitoring scripts and wrappers to distinguish between error and normal exit modes. Failure management of jobs executed as web services or on a grid may require the development of additional tools.

**Hardware requirements** (Joel)

$100,000 in total

8-node (8-core each) cluster

HPC cluster time

Production server

Development server

**Timeline and Milestones:** Provide a timeline for the proposed research indicating points where intermediate objectives will be assessed and decisions will be made regarding the course and direction of the continuing research effort. Possible alternative paths that may be followed at critical junctures in the project plan should be described and indicated on the timeline. Preliminary data are not required but may be included, if necessary to demonstrate the feasibility of the proposed studies. The presentation must be clear and particularly compelling. No detailed scientific plan should be provided, but timelines must be presented.

In addition to the formal software design document detailed above, the project uses an iterative/incremental development model, starting with a simple implementation of the basic functions (workflow, data management, reporting and genome view) and iteratively enhancing at each build with design modifications and new functional capabilities until the release version (Figure 6). Each iteration includes an examination of both the functional and quality requirements, the latter defined by user feedback and user testing (49) and interaction with other stakeholders. The alpha version will be used by a selected group of ‘power users’ within the University of Miami and the source code will be freely available for academic users. For the beta release, the user group will be expanded to all NextGen data users within the University of Miami and registered external users. Extensive user testing will be carried out at regular intervals and the results of the tests will be used to inform any additions or changes to the system’s functional requirements.



Figure 6. Project timeline and milestones. Following a 3-month project Inception stage (currently underway), user testing will take place at regular intervals during the Elaboration, Construction and Transition stages.

**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*)

**http://grants.nih.gov/grants/guide/rfa-files/RFA-OD-09-003.htmL**

**Special Instructions for PHS398 Research Plan Component (Section 5.5 of SF424 (R&R) Application)**

**Research Plan:** The Research Plan is comprised of special sections noted below and is limited to a total of **12** pages, including tables, graphs, figures, diagrams, and charts. The Research Plan should be self-contained and uploaded as a single attachment in the Research Designs and Methods item.

**PHS398 Research Plan Component Sections**

|  |  |
| --- | --- |
| **Item Number and Title** | **Instructions** |
| 1. Introduction to Application | Omit (N/A: Resubmissions and Revisions not allowable) |
| 2. Specific Aims | One page maximum. Separate PDF attachment |
| 3. Background and Significance | Omit |
| 4. Preliminary Studies/Progress Report | Omit |
| 5. Research Design and Methods | Item 5 consists of the following 4 elements and is limited to 12 pages: A statement of the Challenge Area and specific Challenge Topic; The Challenge and Potential Impact; The Approach; and Timeline and Milestones. Attach the 12- page Research Plan encompassing all of these elements as a single PDF document. Figures and illustrations may be included but must fit within the 12-page limit. Do not include links to Web sites for further information. Do not include animations. |

Excluded from the 12-page Research Plan limit are the following items:

* Specific Aims (1 page maximum)
* Inclusion Enrollment Report
* Protection of Human Subjects
* Inclusion of Women and Minorities
* Targeted/Planned Enrollment
* Inclusion of Children
* Vertebrate Animals
* Select Agent Research
* MPI Leadership Plan
* Consortium/ Contractual Arrangements
* Letters of Support
* Resource Sharing Plans

Note the 12-page limit also excludes the Project Summary/Abstract; Bibliography and Literature Cited; and Biographical Sketches (separate PDFs).

Organize the Research Plan in the specified order using the instructions provided below. Start each section with the appropriate section heading (i.e., Statement of the Challenge Area and the specific Challenge Topic, The Challenge and Potential Impact, The Approach, Timeline and Milestones.)

**Research Area:** State which broad Challenge Area (e.g., (01: Behavior, Behavioral Change, and Prevention) described within this FOA and specific Challenge Topic (e.g., *Mechanisms of Behavior Change Research*: *01-GM-104*) will be addressed. Also include the project title on the first page.

**The Challenge and Potential Impact:** What is the research opportunity, scientific knowledge gap or technology that will be addressed? How broad is the potential impact in science and/or health? Which community (ies) will be affected? What is (are) the size(s) of the community(ies)? Will the potential impact be major?

**The Approach:** How will you attempt to explore or solve the stated research problem? How will your rationale and/or approach overcome existing challenges or barriers in the field? If you propose to improve existing technologies or to develop new technologies, which needs are being addressed and what is unconventional and exceptionally innovative about your approach? Provide enough information for reviewers to determine what you are proposing to do, but do not include a detailed experimental plan.

**Timeline and Milestones:** Provide a timeline for the proposed research indicating points where intermediate objectives will be assessed and decisions will be made regarding the course and direction of the continuing research effort. Possible alternative paths that may be followed at critical junctures in the project plan should be described and indicated on the timeline.

Preliminary data are not required but may be included, if necessary to demonstrate the feasibility of the proposed studies. The presentation must be clear and particularly compelling. No detailed scientific plan should be provided, but timelines must be presented.

**Inclusion of Women, Minorities, and Children in Challenge Grant Studies**

For Challenge Grant applications that propose human subjects research, applicants are expected to set forth sex/gender-based hypotheses and plans for data analysis based on a consideration of the relevant literature if the proposed study has the potential for such consideration. The purpose of this approach is three-fold: to ensure compliance with the NIH Guidelines for Inclusion of Women and Minorities in Clinical Research; to capitalize on the growing body of research demonstrating sex/gender differences in all areas of NIH research from basic to clinical and translational; and to ensure that any sex/gender-specific solutions/answers to the stubborn questions are not overlooked, thus resulting in incorrect conclusions/generalizations with respect to men or women. If these sex/gender-based hypotheses are not relevant to the proposed research, applicants should provide scientific justification for why sex/gender analysis would not be relevant.

Applicants for Challenge Grants are expected to address the inclusion of members of minority groups and their subpopulations in developing a research design appropriate to the scientific objectives of the study and set forth racial/ethnic-based hypotheses and plans for data analyses based on a consideration of the relevant literature.

The purpose of this approach is to: 1) ensure compliance with the NIH Guidelines for Inclusion of Women and Minorities in Clinical Research; 2) address gaps in what is known about health disparities between racial/ethnic groups; and 3) ensure that any potential answers to stubborn questions are not overlooked, thus resulting in incorrect conclusions and/or generalizations. If the inclusion of members of minority groups and their subpopulations is not relevant to the proposed research, applicants should provide scientific justification for why racial/ethnic analyses would not be relevant.

Applicants for Challenge Grants that include children are expected, consistent with the "[NIH Policy and Guidelines on the Inclusion of Children as Participants in Research Involving Human Subjects](http://grants.nih.gov/grants/guide/notice-files/not98-024.html)," to set forth age-appropriate hypotheses and plans for data analyses based on a consideration of the relevant literature. This approach is designed: 1) to promote better compliance with the NIH Pediatric Inclusion policy; 2) to address wide gaps in what is known about clinically significant differences, between children and adults and among children of different ages and developmental stages, in the diagnosis and treatment of diseases and conditions; and 3) to ensure that any potential answers to stubborn questions in pediatrics, as well as in early origins of adult disease, are not overlooked. If age-appropriate hypotheses are not relevant to the proposed research, applicants should provide a specific, scientific justification for why age-appropriate analyses would not be relevant.

**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. 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.

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

13. 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.

14. 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.

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

16. 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.

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

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

19. '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-.

20. 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.

21. 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.

22. 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.

23. 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.

24. 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.

25. 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.

26. 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.

27. 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.

28. 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.

29. Stein L. Creating a bioinformatics nation. Nature. 2002;417(6885):119-20.

30. Bare JC, Shannon PT, Schmid AK, Baliga NS. The Firegoose: two-way integration of diverse data from different bioinformatics web resources with desktop applications. BMC Bioinformatics. 2007;8:456. PMCID: 2211326.

31. Gordon PMK, Trinh Q, Sensen CW. Semantic Web Service provision: a realistic framework for Bioinformatics programmers. Bioinformatics. 2007;23(9):1178-80.

32. Oinn T, Addis M, Ferris J, Marvin D, Greenwood M, Carver T, et al. Taverna: A tool for the composition and enactment of bioinformatics workflows. Bioinformatics. 2004;20(7):3045 - 54.

33. Baker PG, Brass A, Bechhofer S, Goble C, Paton N, Stevens R. TAMBIS--Transparent Access to Multiple Bioinformatics Information Sources. Proc Int Conf Intell Syst Mol Biol. 1998;6:25-34.

34. Baker PG, Goble CA, Bechhofer S, Paton NW, Stevens R, Brass A. An ontology for bioinformatics applications. Bioinformatics. 1999;15(6):510-20.

35. Stevens R, Baker P, Bechhofer S, Ng G, Jacoby A, Paton NW, et al. TAMBIS: transparent access to multiple bioinformatics information sources. Bioinformatics. 2000;16(2):184-5.

36. Rak R, Kurgan L, Reformat M. xGENIA: A comprehensive OWL ontology based on the GENIA corpus. Bioinformation. 2007;1(9):360-2. PMCID: 1891717.

37. Karolchik D, Baertsch R, Diekhans M, Furey TS, Hinrichs A, Lu YT, et al. The UCSC Genome Browser Database. Nucleic Acids Res. 2003;31(1):51-4. PMCID: 165576.

38. Kuhn RM, Karolchik D, Zweig AS, Wang T, Smith KE, Rosenbloom KR, et al. The UCSC Genome Browser Database: update 2009. Nucleic Acids Res. 2009;37(Database issue):D755-61.

39. Stalker J, Gibbins B, Meidl P, Smith J, Spooner W, Hotz HR, et al. The Ensembl Web site: mechanics of a genome browser. Genome Res. 2004;14(5):951-5. PMCID: 479125.

40. Stein LD, Mungall C, Shu S, Caudy M, Mangone M, Day A, et al. The generic genome browser: a building block for a model organism system database. Genome Res. 2002;12(10):1599-610. PMCID: 187535.

41. Donlin MJ. Using the Generic Genome Browser (GBrowse). Curr Protoc Bioinformatics. 2007;Chapter 9:Unit 9

42. Lewis SE, Searle SM, Harris N, Gibson M, Lyer V, Richter J, et al. Apollo: a sequence annotation editor. Genome Biol. 2002;3(12):RESEARCH0082. PMCID: 151184.

43. Misra S, Harris N. Using Apollo to browse and edit genome annotations. Curr Protoc Bioinformatics. 2006;Chapter 9:Unit 9 5.

44. Simmhan Y, Plale B, Gannon D. A survey of data provenance in e-science. SIGMOD Record. 2005;34(3):31 - 6.

45. Feagan L, Rohrer J, Garrett A, Amthauer H, Komp E, Johnson D, et al. Bioinformatics process management: information flow via a computational journal. Source Code for Biology and Medicine. 2007;2(1):9.

46. Salzberg S. Genome re-annotation: a wiki solution? Genome Biology. 2007;8(1):102.

47. Shah S, He D, Sawkins J, Druce J, Quon G, Lett D, et al. Pegasys: software for executing and integrating analyses of biological sequences. BMC Bioinformatics. 2004;5(1):40.

48. Garcia Castro A, Thoraval S, Garcia LJ, Ragan MA. Workflows in bioinformatics: meta-analysis and prototype implementation of a workflow generator. BMC Bioinformatics. 2005;6:87. PMCID: 1090554.

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