

# **Workshop Report: Petascale Computing in the Biological Sciences**

**Edited by: Allan Snively, Gwen Jacobs, and David A. Bader**

### **Acknowledgements**

The editors would like to thank Manfred Zorn and Christopher Greer of the National Science Foundation's Division of Biological Infrastructure for their generous sponsorship of this workshop. We thank our colleagues, the workshop attendees (who are listed at the end of this document) for their participation in the active discussions and debates at the workshop, as well as their thinking, and oral and written contributions, around which this report is constructed. Special thanks are due to the working group leads: Ananth Grama of Purdue, Kim Baldrige of University of Zurich, Olivier Lichtarge of the Baylor College of Medicine, and Chris Johnson of the University of Utah, who led the discussions in each working group while capturing as much input as possible in real-time, and then iterated with the editors to sharpen and expand the resulting recommendations and findings.

## Introduction

In view of The National Science Foundation's recent announcement entitled: Leadership-Class System Acquisition - Creating a Petascale Computing Environment for Science and Engineering, which calls for deployment of a petascale computational facility capable of sustained scientific applications performance approaching a petaflop ( $10^{15}$  floating point operations per second) to solve scientific questions of strategic importance by or around the year 2011, and because the time between now (2006) and then is not more than ample for scientists to prepare to use such a computer, a workshop entitled Petascale Computing in the Biological Sciences was organized and held at NSF headquarters on August 29<sup>th</sup> and 30<sup>th</sup>, 2006. The objectives of this workshop were to examine the opportunities for progress in the Biological Sciences that could be enabled by the petascale computational capability and to determine the steps necessary to ensure that this community is prepared to take advantage of such resources when they come on line. The workshop report is below, with a concise bullet list of *recommendations* provided first for ease of reference, and the more detailed *analysis and findings* that led to these recommendations follows.

### I. Recommendations

The potential benefits of petascale computing to advance scientific discovery in biology, and to improve the health and welfare of U.S and world citizens, as identified in the analysis and findings below, are manifold. However, to achieve these benefits, several specific steps are required, both of the community of research scientists in biology and computer science that should advance relevant science investigations, and by funding agencies that should provide resources to carry out these science investigations. The recommendations of this workshop, developed from the analysis and findings section are:

- **A portfolio of candidate petascale applications should be established, and development funding should be provided, for collaborative teams of biologists and computer scientists to prepare calculations that can both advance scientific discovery and run at petascale.** As a community we need to assemble a portfolio of application classes on the path to petascale. We should *not* strive to make this a large, comprehensive portfolio at first, but rather seek to obtain high success by enabling a few strategic biology computations at petascale in the first few years of system availability, particularly to solve science problems of impact. This set of workshop participants identify the following set of broad biology application areas as being inherently suitable for development to go to petascale, and have science impact, in the near term (it is not implied this list is exhaustive). Thus candidate applications may be considered from the following broad areas:
  - **Biomolecular Structure Modeling** (for example extending classical Molecular Dynamics calculations to account for quantum mechanical effects, multidimensional free energy surfaces, transition state ensembles)
  - **Modeling Complex Biological Systems** (for example developing models of cell and organ function)
  - **Genomics** (for example search calculations mapping phylogeny to ontogeny)

- **Customized Patient Care** (for example computing drug interactions in the context of individual physiology and blood chemistry)
- **Ecological component of earth system modeling** (for example adding plant cover to climate models)
- **Infectious disease modeling** (for example modeling of disease spreading and the likely impact of containment strategies)

The reasoning used to identify these broad areas as suitable for petascale, as well *specific instances of calculations from each suitable for petascale* are further described below in the analysis and findings section.

- Explicitly called out from the above bullet: **projects to scale candidate biology calculations to petascale should be undertaken by collaborative teams of biologists and computer scientists.** All the calculations identified in the analysis and findings section, even the relatively straight-forward ones, require work by both experts in the domain science and in the associated computer science, working together, to get ready for petascale.
- Extending the above beyond the short-term: **petascale community should be cultivated.** To foster a growing interdisciplinary community of collaborating biologists and computer scientists we recommend organization of “summer institutes”, focused workshops for carrying these collaborations forward, and key training programs for next-generation interdisciplinary scientists in this field. These should be advertised widely to all agencies with funding/interest in computational biology including (beyond NSF) NIH, DOE, DARPA. Petascale application proposals should also be encouraged to propose “High-Performance Computational Biology postdocs”.
- **Suitable selection criteria should be applied in choosing which specific calculations and teams from the above areas to invest effort and money into for the purpose of enabling viable petascale applications (via several years of effort starting ASAP).**

We propose the following broad evaluation criteria:

- Reward due to science impact
- Strength and interdisciplinary nature of proposed team
- Demonstrated plausibility for petascale (via reports from team initial feasibility studies)
- Needed investment in algorithms and models (via reports from team initial feasibility studies)
- Risk of failing to result in a viable petascale application

We applied the first and last (risk versus reward) in identifying the broad application areas suitable for petascale in the first bullet. Further effort is required to evaluate teams and carry out feasibility studies—this further effort should be supported by NSF and other relevant agencies by appropriate means.

- Explicit from the above **initial feasibility studies should be carried out ASAP.** Collaborative teams should be formed to further assess the suitability of calculations, including from the broad areas of the first bullet, to achieve petascale. Performance studies should be generated showing where scaling bottlenecks are and strategies for working around these should be explored. To mitigate risk of failure, teams should not receive larger multi-year development funding before justifying further advancement via short (1 year or less) feasibility studies.
- **Innovative uses of a petascale computer should be cultivated.** There is a danger that some other applications of petascale computing that are untested, but innovative and with potentially very high impact will be squeezed out early by the approach embodied in the above recommendations. To mitigate this danger we propose an additional specialized RFP with modest funding for assessment and development of “shallow end of the pool” research that is “High Reward/High Risk” and “High Reward/High Effort)
- **“Market segmentation” should be done.** We do not believe it is the case all computational problems in biology are potentially petascale. At the same time, many biology applications may stress memory bandwidth, disk I/O rates, integer functional units, database query rates, in ways that traditional high performance computing, floating-point intensive (such as physics codes solving systems of PDES (partial differential equations) ) do not. There is a need and opportunity for the community to examine computational requirements of biology applications, and, in addition to identifying candidates for petascale, identify attributes that may be used to influence such programs as NSF OCI (Office of Cyberinfrastructure) “track 2” system procurements. The result could be a machine at less than petascale but well suited to the data-intensive applications of biology.
- **High-performance algorithm development should be supported.** Many problems in biology are irregular in structure, significantly challenging to parallelize, and integer-based, using abstract data structures, all characteristics that are unlike traditional uses of supercomputers for regular, numerical computing. As the biological sciences community is a relatively young user of parallel and high-performance computational resources, adequate support should be provided towards the design and optimization of algorithms for large-scale computational problems in the biological sciences. The result would be advances in algorithms and the early opportunity to include them in emerging petascale applications.
- **Software support and maintenance should be supported.** We emphasize the NSF-wide need to develop a model and paradigm for making software solutions robust, maintainable, and reusable. This likely requires staff support, as opposed to graduate student, software support (students can invent but should not be expected to harden and maintain software.)

*The NSF BIO and CISE directorates need to increase investment in people and software applications development commensurate with the outlay in funding for hardware from OCI to enable petascale computing. We recommend a Biological Petaflop Computing Initiative, on the model of NMI (NSF Middleware Initiative) to foster software development. A focus should be on specialized programs to capture the specific biology subfields for development in the context of petascale computing. There should be an emphasis on coupling and coarse-graining techniques to enable scaled-up coupled models. Very specifically, we recommend support of the scaling of molecular dynamics to the petascale level, but also to support the development of coarse grained techniques that, in themselves might not be petascale, but would add great biological discovery value to the petascale molecular dynamics calculations. There should be an RFP for BIO HPC/parallel algorithm development and this could be in collaboration with (beyond BIO), the NSF CISE and Nano directorates.*

BIO should explore inter or cross agency coordination to address funding for Application Services, libraries, runtime/programming environments needed at petascale. This is a need of biology applications at petascale, but the need is too large and general to be the sole responsibility of BIO to provide.

- **Storage and networking should be supported.** It is recommended that NSF support national infrastructure for data-intensive applications from the biological sciences, including storage and database resources. For example, the CISE and DDDAS (Dynamic Data Driven Applications Systems) directorates and BIO could jointly support a solicitation in this area.

The result of following these recommendations will, we believe, both enable a few early successes of petascale biology applications solving important science problems running on the petascale facility when it first becomes available, and enable the evolution of a balanced, robust, interdisciplinary, computational biological sciences community and infrastructure going forward.

## **II. Analysis and findings**

The workshop was structured around the following broad questions:

1. What are examples of important questions and conceptual challenges in the biological sciences that illustrate the potential impact to the biological sciences of access to a petascale computational facility?
2. What strategies will ensure that the biology community is in position to take advantage of petascale computational capabilities?

3. What resources are needed to get teams of biologists and computer scientists working together on petascale applications development, with the goal of having operational packages ready by 2011 when petascale resources will come on line?

After a plenary session and keynote talks, we formed three parallel working groups to address each question, and then convened a final plenary session to seek consensus among the participants.

The remainder of this report describes the findings of each working group, followed by a summary of recommendations based upon these findings.

## **II.A. Petascale Applications**

This working group was tasked with identifying candidate petascale applications in biology. The focus included critical issues, outstanding challenges, and potential impact, as well as on scaling up existing applications to petascale, and on turning important questions and conceptual challenges into petascale applications.

In the 50 years since the discovery of the structure of DNA, and with new techniques for sequencing the entire genome of organisms, biology is rapidly moving towards a data-intensive, computational science. Many of the newly faced challenges require high-performance computing, either due to the massive-parallelism required by the problem, or the difficult optimization problems that are often combinatoric and NP-hard. Unlike the traditional uses of supercomputers for regular, numerical computing, many problems in biology are irregular in structure, significantly more challenging to parallelize, and integer-based using abstract data structures.

In the large view, some candidate petascale calculations in biology are already extant as applications, some are even running at terascale ( $10^{12}$ ), while others exist only as abstractions, models, and (in some cases) algorithms for solving them. For these latter conceptual problems, implementation is an issue; the quality of the implementation may be a larger factor than intrinsic suitability, or scientific importance, in determining their success at petascale. Even for the former cases (existing codes), there is no simple proof that either a) terascale applications will naturally scale to petascale, or that b) important science questions would be answered by so doing; rather, the suitability of each for scientific importance, scalability, and computational challenges, must be examined case by case. Still, in any case, validation is an issue. In many computational science problems, determining that a calculation is computing a result of high fidelity to nature and of scientific relevance, will be as or more difficult than implementing the application at petascale to begin with.

With the above larger issues in mind, three guiding principles were used by this group to identify possible candidate petascale applications in the biosciences: 1) needs of the domain science are more important than simply enabling petaflop calculations 2) people are expensive, machines are cheap (or in other words software is expensive and hardware

is relatively cheap), so designing, coding, porting, tuning, and validating applications will be at least as expensive as procuring petascale hardware, and as well are of utmost importance in this drive towards petascale computing. Furthermore, a rather ideological position was taken by this working group, that being that 3) “nothing in biology scales to a petaflop, unless otherwise demonstrated or proven.”

With these guiding principles in mind, candidate questions and conceptual challenges were identified and deemed potentially suited for petascale within the available time, given sufficient resources, early start, and ample time for success. These questions are listed below in rough order of deemed readiness/nearness deployment at petascale, with the most mature candidates listed first. In addition, for each category of application, some risk/reward assessment is provided. Risk is loosely defined here as risk of either 1) failing to be deployed at petascale (i.e. due to difficulty of implementation) within the timeframe, or 2) failing to compute a result of significant scientific merit, or both. Reward is of course the opposite i.e. 1) likelihood of running at petascale at “first light” if sufficient resources are devoted to software development, and 2) importance of the underlying problem to biologists, or both.

- **Biomolecular Modeling (from structure to function)**

There is an opportunity for a petascale computational facility to enable calculations that extend current computational models of molecular structure and dynamics both in dimensions of detail per molecule and in dimensions of number of molecules that interact. Such enhanced detail should make it possible to explore phenomena, for example, where quantum dynamics and thermal motions interact to model accurately the mechanisms for essential biological function, such as electron and proton transfer. This should also allow electronic structure calculations essential for the next stage in force field improvement of classical molecular dynamics. Potentially then, calculations involving many more atoms and molecules interacting than is state-of-the-art today could enable understanding of the assembly and function of larger molecules into complex molecular machines that include protein, RNA and membrane components and that could result in manifold practical benefits to areas such as drug design and understanding of the ageing process, while furthering fundamental understanding of biological processes. However, the path to this opportunity is not quite straightforward.

It is considered relatively low risk to implement and scale an existing classical molecular dynamics code to petascale; calculations using state-of-the-art codes today already simplify their models substantially just to make the computation tractable (e.g., doable on a terascale machine). Therefore, in principle, it is straightforward to add more physics and chemistry to *possibly* increase relevancy of the simulated outcome. This is not to say that, in moving to petascale, that existing methods for calculating, for example, long range electrostatics, will not have to be rethought, possibly posing major algorithmic and software challenges. However, simply extending detailed molecular dynamics in this way may not alone be high reward for biologists in general; in the worst case, one would have the situation where a petascale calculation would be enabled that is not of wide community interest. Today, significant high end

compute time for “biology” is used by the molecular dynamics community, but many biologists consider the results of that work, while interesting, much less significant than results generated from bioinformatics in their subfields. On the other hand, if the atomically detailed classical molecular dynamics could be augmented by coarse-grained models that are parameterized by the molecular dynamics output, extending all the scales of time and length and comprehensiveness of phase space sampling, then the results may become very significant for biology in general, and for physical biochemistry in particular.

A prime class of problems for molecular dynamics/coarse-graining at the petascale is assembly and function of heterogeneous biological membranes. The system sizes that are desirable to explore will challenge even petascale machines, and the biological issues are of major importance. A petascale calculation in this domain would involve systems say larger than one hundred million atoms and would capture how functions may manifest over unprecedented timescales.

From the biological standpoint, studies of anomalies can often be as or more important than the related canonical results (for example protein misfolding is as interesting as protein folding), and could be the focus of early important petascale calculations. Models will out of necessity be multiscale and the implementation thereby typically hierarchical (system ensembles). Such implementations will pose particular computational challenges requiring dynamic load balancing, coarse-graining, dealing with time-serialization, solvent models, electrostatics and the like, to enable efficient execution. Work must begin as soon as possible on these difficult implementation challenges to ensure success within the timeframe.

### **Examples of extending Molecular dynamics calculations to address interesting questions in Biology**

Because many participants at the workshop agreed that large-scale molecular dynamics simulations as described above are the most likely candidates to achieve petascale performance, but that molecular dynamic by itself might not hold as wide of interest to many biologists, here in an expanded subsection, are some specific examples of applications in this domain that could be made to scale on a large fraction of the proposed machine with relatively modest effort for underlying algorithmic modification, while at the same time enabling interesting problems to be addressed by biologists.

#### **Example 1: Quantum mechanics in biology.**

Considering that proteins are composed mostly of light (first or second-row) elements, nuclear quantization effects may be manifested significantly in these systems. The importance of these effects, especially in the context of biological proton transfer, has long been debated in the field of enzyme catalysis. It is an issue that deserves careful investigation; especially in the light of experimental findings (Cha, Y., Murray, C. J. & Klinman, J. P. (1989) *Science* 243, 1325-1330) suggesting that proton transfer in enzymes (e. g. bovine serum amine oxidase) may proceed through tunneling. Progress to date has been hindered by the unavailability of

adequate computational resources. Nuclear quantum effects at finite temperature can be incorporated into ab initio molecular dynamics via the Feynman imaginary time path integral (PI) formulation of quantum statistical mechanics. It is, therefore, possible to investigate the influence of nuclear tunneling and zero point vibrations on finite-temperature equilibrium properties, while maintaining a first-principles description of the system. The PI methodology is well established, however, applications have been limited to proton transfer in bulk liquids, water wires and small water clusters due to the enormous computational requirements of the method. It can be expected that comprehensive path integral investigation for a handful of well-characterized systems (dihydrofolate reductase, carbonic anhydrase, triosephosphate isomerase) would become feasible if a petascale platform became available. For instance a MM/QM (Molecular Mechanics and Quantum Mechanics) model of the dihydrofolate reductase active site comprising 200-300 atoms in the QM subsystem, simulated with a sufficiently large basis set to ensure accuracy ( $\sim 100$  Ry cutoff), could scale (depending on the machine architecture and the chosen parallelization scheme) from several hundred to several thousand processors. The PIMD method maps the problem of a quantum particle into one of a classical ring polymer with beads that interact through spring forces. The bead discretization entails simulation of multiple replicas of the system and requires only very limited communication. Simulating 32 to 64 beads (to ensure convergence) would, thus, result in utilization of  $\sim 64,000$  processors. This firmly positions such an application, if not in the petaflop, at least in the hundred-teraflop regime. The results could be of potentially great importance, as they would allow researchers to evaluate the applicability of a posteriori corrections for nuclear quantization, widely applied in computational work on enzyme catalysis today. (Note: The QM/MM module of Car-Parrinello molecular dynamics (CPMD) at present is not fully compatible with the PIMD options. However, there is no conceptual obstacle to implementing this in the code.)

***Example 2: Multidimensional free energy surfaces (FES) for enzymatic catalysis.***

As an attractive alternative to more established sampling methods such as blue moon ensemble/constrained dynamics or umbrella sampling, a very recent development in reactive CPMD has been introduced. The method is based on (i) metadynamics in the phase space defined by a reduced set of collective coordinates, (ii) coupling of the metadynamics to the real dynamics of the system through the Car Parrinello Lagrangian and (iii) the application of an adaptive biasing potential. In this approach, the reaction coordinate is discouraged to remain fluctuating around the local minimum of the reactant state, by adding a small Gaussian-shaped potential every few hundred dynamics steps. In this way, the reactant free energy well will be completely filled with Gaussians during the molecular dynamics run, allowing the system to escape the minimum via the lowest transition state to the next (product or intermediate) well. The free energy surface of the system, including the transition state regions, can be reconstructed from the sum of the accumulated Gaussians. The ability to trace independently more than one or two collective degrees of freedom allows for the study of concerted reactions – a capability, which is of great importance in the investigation of enzyme mechanisms. Furthermore, the method can

be generalized to multiple-walker metadynamics. Just as in the previous example, the scalability of such computational scheme makes the method interesting from the point of view of potential petascale applications. The computational cost of sampling increases exponentially with the number of collective reaction coordinates making determination of multi-dimensional free energy surfaces in enzymatic catalysis an exceptionally challenging problem. To illustrate the point consider the enzyme uracyl DNA glycosylase (UDG) – a vital DNA repair protein, key for the maintenance of genomic integrity. It has been long debated whether the reaction mechanism is associative or dissociative. To distinguish between these mechanistic alternatives, at least two independent coordinates need to be considered (and a two-dimensional FES evaluated). Furthermore, if the role of initial strain in the active site needs to be addressed, which has been suggested as mechanistically important, one would have to consider the out-of-plane motion of the deoxyribose ring and add an extra coordinate (two-dimensional FES). Thus, increasing the dimensionality of the sampled phase space allows one to answer increasingly more complex questions about the system. As in the previous example, since a QM/MM model of the enzyme is expected to scale from 300 to more than 1000 processors and 100 independent walkers can be used to sample exhaustively the region of phase space of interest, it is likely that such an application could make efficient use of  $\sim 10^4$  -  $10^5$  processors.

***Example 3: Transition path sampling applied to enzymatic catalysis and determination of transition state ensembles.***

Recent experimental work has unequivocally demonstrated the dynamic nature of enzyme catalysis at the single molecule level, leading to a distribution of free energy barriers for individual catalytic events. Therefore, to further understanding of catalytic processes, it is important to go beyond the single transition state picture. A more accurate description may be achieved by considering an ensemble of transition states evolving on a free energy landscape, dynamically transformed through conformational changes occurring on a variety of timescales. In this context, it may be more appropriate to describe the reactive processes in terms of ensembles of states associated with the transition region and reactant and product basins on the free energy surface. Such detailed description, based on the concept of dynamical path sampling, has been pioneered by Chandler and coworkers. The method, called transition path sampling (TPS), essentially amounts to importance sampling in the space of individual trajectories. A recent application to the problem of water autodissociation (Geissler, P. L., Dellago, C., Chandler, D., Hutter, J. & Parrinello, M. (2001) *Science* 291, 2121-2124) highlighted the extreme computational cost of transition path sampling. Using transition path sampling techniques to investigate the detailed mechanism and kinetics for select reaction steps in an enzymatic catalytic cycle would require orders of magnitude larger computational effort. In essence, the method consists of: (i) selecting an initial guess for a trajectory connecting the reactant and products states; (ii) generating new trajectories by randomly shifting the momenta (shooting moves) at an intermediate point of the initial path; (iii) acceptance or rejection of the newly generated paths based on a Metropolis criterion; (iv) repetition of the above steps until sufficient sampling is achieved. Sampling in trajectory space is especially computationally intensive due to the need to calculate

multiple trajectories. Coupled to the significant computational expense of CPMD, problems proposed in such calculations can be classed as exceptionally difficult. To achieve an accurate representation of the TS ensemble for an enzymatic reaction, a few hundred trajectories may be needed. Assuming an ideal acceptance ratio of 0.5, up to 1000 short (1 to 2 ps) trajectories may have to be integrated forward and back in time bringing the potential processor count for the application once again into the 10<sup>4</sup>-10<sup>5</sup> range. (Multiple trajectories may be computed simultaneously with no communication overhead).

The key unifying feature of these examples, besides their focus on important biological problems, is the exploitation of multiple levels of parallelism, coupled to the large computational requirements of the *ab initio* molecular dynamics method. The support of such calculations appears relatively low risk for achieving a petascale calculation in the appropriate timeframe and also relatively high reward for addressing interesting biology problems.

- **Modeling complex biological systems—From Cells to Organs (neuronal modeling, cardiovascular simulations, radiation transport, pathophysiology)**

At the next higher level in model size and complexity an opportunity exists for a petascale computational facility to enable modeling of unprecedented, large, macromolecular constructs at the cells and organs (e.g., the heart) level. Such focus is considered by many to be the next level of grand challenge problems in computational biology, with many possible benefits to health and welfare, to say nothing of the ability to deepen our understanding of human and other life-form bodies. There is indeed little difficulty in actually posing such large calculations (even beyond petascale), however, there is substantially more risk for implementation, relative to the former bullet, particularly since the models are more complex to implement, not yet well established beyond limited cases, and the fundamental biology principles less well understood at this level. Petascale computations in this domain, for example a model of the human heart, could involve finite elements, possibly 10<sup>10</sup> higher order, and would involve hybrid/stochastic computational methods.

Likely, implementations would include multiscale/multiphysics models, Monte Carlo methods, reaction-diffusion in both intracellular and extracellular space, and/or circuit models. Additionally, in many applications, geometry is a major consideration. Due to the complexity and hybrid nature of these calculations, load-balancing and coarse graining will be even more difficult than described in the previous class above.

As an example application: multiscale modeling of cardiac arrhythmia - from ion transport to the organ level. At the organ level, one would need about 10<sup>8</sup> to 10<sup>9</sup> degrees of freedom. This includes mixed domains, i.e. flow, structural and electrophysiology. One would also need long-term integration, about 100 cardiac cycles with 15,000 time steps. Based on current computational indices for such simulations, one can arrive at least a week of petascale computing to model an incident of arrhythmia. If one extends the model to include details of cellular

dynamics (as for example in the MCELL code of Joel Stiles) and also NAMD type simulations for subcellular activity, then such calculations could require several weeks per episode modeled.

Moving forward with calculations such as these, involving an organ performing some interesting “behavior”, and supporting that for further work to deploy at petascale, would be considered relatively medium risk for achieving a petascale calculation in the timeframe, as well as relatively medium reward for addressing interesting biology problems (lower reward because it could result in a “home run” for example by deepening understanding of the organ and potential for malfunctions, but also has the potential to result in a large scale calculation for low relevancy).

- **Genomics**

Genomics is a broad and growing area in biology. Only some of the associated calculations are potentially petascale however. We consider some that have potential in the following bullet list:

- **Genome and Metagenomic Correlations** involve calculations on, for example, genomes of entire microbial communities. The science impact of such computations are somewhat debated within the biology community; however, they could be of very high reward if such computations pay off strongly in biological understanding. Such abilities lend themselves towards biological systems-level understanding. As well, such computations are, in principle, highly scalable, since they are based fundamentally on pairwise correlation, and thus representable by matrices. The challenge is in the data-intensiveness of the computation, and the size of the data stores, as exemplified by Genbank, is still doubling about every 18 months. This is both a justification as well as a challenge for a petascale computing facility. Data intensity handled in a brute force way requires large memory, but memory is expensive and not becoming more available on the same curve as processing or the size of biological data stores. It is possible that the petascale facility will have no more memory *per processor* than today’s machines, but simply more processors. The software engineering challenge is to handle memory more efficiently for data intensive applications—the hardware challenge is to provide sufficient globally addressable memory to permit scaling of such applications. These challenges are inherent in going to petascale and not unique to biology applications.
- **Genomics in the small** includes possibly petascale calculations involving, for example, genotype Single Nucleotide Polymorphisms (SNP)-phenotype modeling (50M SNPs x 1000 individuals x 50 phenotypes). What is unclear is whether or not such computations are in fact petascale, or instead require only terascale (or smaller) caliber resources. Further assessment is required to fully understand the computational size of these types of problems. Whole genome association studies, at the large-scale

end, are currently about 300K SNPs, with 1K cases and 1K controls (NIH IC Director's Meeting Highlights, 28 April 2006). Even at 3K X 2K, the problem is not very large. Although the number of genotype markers may increase to something on the order of 3M, and the number of individuals phenotyped may increase on the order of 100K, the problem may not be of sufficient scale to warrant petascale resources. Furthermore, some of the current top statistical and machine learning algorithms take advantage of randomness to decrease variance and increase accuracy, primarily through re-sampling and sub-sampling techniques. These algorithmic techniques not only improve analytical results, but reduce computational requirements (e.g., memory and processors).

Supporting development of techniques such as these to scale up genomics for deployment at petascale, appears relatively high risk for achieving success in the given time frame. It is not clear that the required globally addressable memory, or the associated data handling infrastructure required, much less the data mining algorithms to support it, would be available on the petascale facility at "first light." Although such computations are of potentially high reward for addressing interesting biological problems, and appropriate for petascale facilities in terms of their ability to stress infrastructure for data, it is not fully evident that this area can fully exploit petascale architectures. It may indeed need a different kind of architecture that better supports data intensive applications.

- **Genome-wide association (cell division, genotype-phenotype association)** provides an opportunity for a petascale computational facility to enable unprecedented, deep and wide, mining of genotype and phenotype data to establish association. Generation of massive amounts of genomic data is at an unprecedented level, however, the implications and significance of genetic codes is mostly still unknown. The possibilities for petascale computation to enable a renaissance of biocomputing by enabling data-mining at a scale required to map genomic codes to organism physiology and behaviors, is still unclear. A candidate calculation would follow a data, to model, to knowledge, cycle of development. However, algorithms and models in this arena are ill-understood at present, and, working through issues of data federation and availability would be critical to success. So while the sizes of interesting problems in this domain are clearly petascale or larger, further assessment is required to ensure that concrete and tractable implementations can be arrived at within the given time frame. The solution space for even moderate phylogenetic reconstruction problems is extremely large (i.e., making even petascale seem extremely small). However, the computational bottlenecks using some of the latest best programs for large problems (e.g., GARLI, RAxML), provide some obvious places where petascale computing could be of significant benefit. Current best analytical

approaches in this domain use maximum likelihood or Bayesian methods. The computational step that dominates the processing required for these methods is calculation of likelihood values. These likelihood values can be done in parallel (and several existing programs do this in a limited way). Petascale computing could greatly increase the opportunity for parallelization of the likelihood calculations. Other algorithmic approaches shown to be advantageous in current practice are genetic algorithms and Markov chain Monte Carlo. Although current implementations of these methods would have to be changed and adapted to the purpose, both are intrinsically parallel and so could potentially be significantly scaled up to enable petascale likelihood calculations. Algorithms for solving problems from computational biology often require parallel processing techniques due to the data- and compute-intensive nature of the computations. Many problems use polynomial time algorithms (e.g., all-to-all comparisons) but have long running times due to the large number of items in the input; for example, the assembly of an entire genome or the all-to-all comparison of gene sequence data. Other problems are compute-intensive due to their inherent algorithmic complexity, such as protein folding and reconstructing evolutionary histories from molecular data; some of these are known to be NP-hard (or harder) in finding an exact solution, and they often require approximations that are still algorithmically complex. Phylogenetic analysis attempts to reconstruct, from data generated from extant species, the evolutionary history of the group under study. Because phylogenies are crucial to answering fundamental questions in biomolecular evolution and practical ones in molecular biology, biologists have a strong interest in algorithms that enable resolution of ancient relationships. The reconstruction of large phylogenies could yield fundamental new insights into the process of evolution. The history is generally represented by a bifurcating (binary) tree, a phylogeny.

Choosing calculations that correlate genotype to phenotype, and supporting the necessary further work to deploy the methods at petascale, appears relatively high risk for achieving petascale level in the given time frame. It is not clear that the associated data handling infrastructure required, much less the data mining algorithms to support it, would be available on the petascale facility at “first light”. However, if successful, such calculations would be relatively high reward for addressing relevant biology problems, and indeed seem to be the next wave in computational biology.

As to genomics generally, the explosion of genomic data from annotated gene sequences to whole genome assemblies, combined with mature algorithmic techniques for problems in comparative genomics, will drive computational needs even higher. Hence, high-performance computing techniques will become increasingly important for solving biologically-meaningful problems with realistic problem instances.

As an illustration, Bader et al. have designed and implemented a high-performance software suite, GRAPPA (Genome Rearrangement Analysis through Parsimony and other Phylogenetic Algorithms), with the goal to re-implement, extend, and especially speed up, the breakpoint analysis (BPAnalysis) method of Sankoff and Blanchette. In a recent analysis of a set of chloroplast genomes for 12 species of Campanulaceae on a 512-processor Linux supercluster, GRAPPA achieved a one-million-fold speedup over BPAnalysis—a speed-up that they have since increased to over one billion. This speedup was achieved through a combination of parallelism and high-performance algorithm engineering. Although such spectacular speedups will not always be realized, we suggest that many algorithmic approaches now in use in the biological, pharmaceutical, and medical communities can benefit tremendously from application of high-performance strategies and associated hardware platforms. Thus the risk of investment in petascale genomics application development is perhaps mitigated by the potentially high reward in practical applicability of results.

- **Customized patient care**

It is not too early to imagine that petascale computing may eventually support on-the-fly individualized patient care, including calculations of optimal drug choice and dosage taking into account the interactions and reactions expected from the patient's unique polymorphisms, as read from his or her "DNA fingerprint". Likewise, comparisons of a patient's tomogram or angiogram to a pre-recorded database of normal variants versus pathological cases are other examples of customized healthcare based on the parallel processing and integration of vast amounts of data.

An example would involve solving inverse problems over multimodal imaging data (functional, structural). It would have real-time constraints and this in conjunction with resolution requirements may conspire to make such problems petascale. Additionally we expect the emergence of more expensive covariance models. This is considered a "novel" potential use of a petascale computer and breaks many of the traditional assumptions about how such a system should be deployed.

Risk assessment for such an application area is difficult. There are clearly such calculations that could be sped up to enable soft real-time decision making. However, it is unclear whether or not the National Science Foundation would or should consider "on-demand" scheduling of the petascale facility for such a use.

- **Ecological component of earth system modeling**

There is an opportunity to use a petascale facility to enable adding ecological information, such as forest growth, to models of weather and climate. A grand challenge in geoscience is the addition of clouds to ecological modeling. However, clouds and cloud-formation processes interact (in both directions) with the ecosystem. Ecological models attached to a high-resolution model of climate change (or ocean circulation, etc) that already has clear petascale applicability on its own may be a good path forward. It is debatable whether ecologists will have much say in the

design or deployment of these models, unless they begin to collaborate with the climate modelers as soon as possible. Ecologists have several embarrassingly parallel applications that can be “easily” scaled to a petascale system, including:

- stochastic processes that need replication
- parameter sensitivity analysis
- heuristic optimization

Such problems are extremely common in ecological application, as we elaborate here.

1. Stochasticity: Simulation-based ecological models often incorporate demographic stochasticity (random birth/death/movement, etc), environmental stochasticity (random components of climate forcing, resource availability, etc), and/or genetic stochasticity (random mating, mutation, etc). Outcomes are thus stochastic as well, and ecologists wish to ask questions like, “What is the simulated probability that the population size will fall below X within 100 years?” The simulation model must therefore be independently repeated (usually 100s-1000s of times) to generate a distribution of outcomes.
2. Parameter sensitivity (or more generally, model sensitivity): The “true” parameters of ecological models are rarely known, and in fact there are often disagreements about the form of the equations governing those processes. Consequently, ecologists frequently want to characterize the sensitivity of outcomes to input parameter values and model assumptions. This also requires repeated simulation.
3. Optimization: There are (at least) two distinct types of optimization questions that ecologists commonly ask. The first involves fitting parameters to observed data. In all but the most trivial models, it is impossible to use analytical or even simple approximating techniques to identify maximum likelihood estimates of parameters. Increasingly, ecologists are turning to stochastic optimization techniques such as simulated annealing, or the use of various implementations of Markov Chain Monte Carlo to simulate posterior probability distributions in a Bayesian framework. Secondly, applied ecological models often implement heuristic optimization algorithms as decision tools (e.g. identifying the optimal spatial configuration of a land reserve system, given some cost criterion). As with parameter estimation, the simpler algorithms used in the past have been shown to be deficient in complex settings, but more reliable methods require many repeated simulations and long run-times. There is a tremendous need for HPC solutions that can deliver results sufficiently quickly even for models involving many parameters, fine-scale spatial and temporal resolution, and stochastic processes.

Putting this all together, it is clear that the compute time can be overwhelming when coupling one or more the above procedures with even a moderately complex ecological simulation model. Specifically, some model examples include predicting

evolution of a collection of interacting species, spatial spread of a disease, or the dynamics of a specific ecosystem. Taking the last example, imagine a regional-scale ecosystem model, the core of which is deployed as a small-scale HPC application (e.g. a single simulation that takes days to complete on a cluster with dozens of nodes). Indeed, the ATLSS group (<http://www.atlss.org>) based at UT/ORNL has spent ~10 years developing and refining a model that integrates a variety of complex and interacting submodels to simulate key biological and environmental components of the Florida Everglades; AFAIK, one submodel has already been parallelized to run on 60+ nodes. Even if a researcher demands just several hundred stochastic replications in such a simulation, performed for each of 100 possible configurations of a proposed reserve system, there would be significant benefit from hierarchically parallelization, to enable a 100k-processor system run (imagine a multi-hundred simultaneous, distributed instantiation of the ecosystem simulation, which itself might be a 64-node data-parallel application). Whether the envisioned petascale system even provides the right architecture for this application could be debated, but the point is that it does not require significant effort to scale up moderately sized ecological models to result in large computational needs, resulting in the ability to address relevant and interesting problems.

Here, as in genomics, data integration would be critical to success. A candidate calculation would involve evolutionary correlations of networks and functions (phenotypes). In the worst case, such formulations may lead to NP Hard/Complete problems that would remain intractable even at the petascale level. To the extent that ecologists are able to refine mechanistic mathematical models in a way that is increasingly faithful to reality, one could easily conceive of petascale computing demands for simulating an entire ecosystem from its underlying biological and physical components. However, it is worth pointing out that the tradition in ecology is to simplify and scale back models—indeed, to err on the side of oversimplification; “realistic” models have long been mistrusted in favor of either highly abstract mechanistic (theoretical) models and/or simple phenomenological (statistical) models. In part this is for good reason: ecologists do not yet fully rely on their own more detailed mechanistic models (there being a lack of the ecological equivalents of physical laws, testing approximating models via experimentation and observation is difficult, and each real system seems to have its own unique features). This could in fact partly be historical artifact: few ecologists are even aware of the computational possibilities now afforded by HPC systems. In a sense, one might argue that developments in this area are limited due to apparent belief in computational obstacles that no longer exist—something we believe can be remedied through education and workshop opportunities. Given opportunities for making forays into developing complex ecological simulations despite uncertainties about the models, and having the ability to enhance model output with observed data, has the potential to lead to refinement and progress in this area.

Choosing applications focused on the coupling of ecological with weather and climate models, and supporting them for further work for deployment at the petascale level, appears to be relatively medium risk for success in the given time frame. Much

of the risk is dependent on collocating two disciplines that normally do not interact collaboratively, to facilitate synergistic code co-development. Computational possibilities in this area have the potential for relatively high reward towards addressing interesting biology problems. The path towards petascale is relatively straight forward, and the science questions of great relevance for better understanding of our environment by the ecological communities.

- **Stochastic agent-based population simulation for such areas as epidemiology dynamics and ecosystem dynamics**

A representative calculation in this area would involve agent based modeling of populations, applied to ecology as well as epidemiology, spatio-temporal, genotype/phenotype evolution. Such calculations could greatly enhance decision making for CDC/DHS. Further work is necessary to establish clear ability for such applications to go beyond the terascale level. However, it is almost certain that simulations can be scaled to petascale with some effort, and that the associated science to be gleaned would result in significant new insights into system behavior. Petascale computation could enable multiple computations to generate result ensembles and parameter sweeps for very detailed representation of large-scale systems. In the case of epidemiology, models may be used for real time response guidance to emerging challenges such as a pandemic or bioterror attack, where speed of execution of detailed models would be essential for effective response. In the case of ecosystem dynamics, petascale computing could lead to an effective coupling of biology to factors in comprehensive global change models, an essential next step.

This area is deemed high risk, as there are very limited calculations of this type even at the terascale level today. However, the possibilities offer potentially high reward for answering science questions of national strategic importance.

### **Models and Abstractions**

These example of candidate applications cut across several levels of model granularity and abstraction. As broad model taxonomy, some calculations such as molecular dynamics could be termed *Classical Mechanics Models* (CMM). Others, such as models of complex biological systems could be categorized as MM (*Macroscopic Models* including continuum, and coarse grained approaches), while genomic and ecological models could be termed DDM (*Data Derived Models*, following a Data to Knowledge pattern). A general classification of the above candidate applications, by this taxonomy, would be as follows:

- Biomolecular Modeling as (CMM/MM)
- Modeling Complex Biological Systems as (MM)
- Genomics as (DDM)
- Customized Patient Care as (DDM)
- Ecological component of earth system modeling as (CMMDDM)
- Infectious disease modeling as (DDM)

The taxonomy will be useful for distinguishing infrastructure needs and for planning (next sections).

In many of the problem classes just discussed, models either do not exist or are rudimentary, so that there is a clear need for additional research and development. For some other cases, coarse grained models need to be derived from existing fine grained models to reach the requisite scales. Thus there is a need and opportunity to apply maturing coarse graining technique (elaborated more in next section). Many applications integrate a range of model scales and phenomena; for these, methods for coupling models in efficient and accurate ways will be a crucial enabling technology. Classical examples are in DFT(Density Function Theory)/MD/Continuum/System. Petascale computing will enable scaling up science models from molecules to organisms to ecosystems, from ion transport to cardio-vascular simulation, and the like. There are issues of information flow across models and integrating data into models (model inference) that will manifest at this level of model size and complexity; these details need to be worked through in advance of deployment. The candidate applications discussed above have many algorithm and implementation issues, discussed next.

## **II.B. Petascale Planning**

This group dealt with strategies to prepare for petascale computing, such as:

- 1) The need for interdisciplinary collaboration
- 2) Selection criteria to determine which interdisciplinary teams to sustain (given limited funds and resources),
- 3) Software challenges on the way to petascale
- 4) Community-wide organizational structures that may foster the long-term needs of petascale computing

### **1. Need for Interdisciplinary Teams**

The petascale funding announcement has stimulated an intense and broad degree of interaction between computational biologists and computer scientists. It is critical that this interaction be further sustained, both by NSF and NIH.

Our goal is to build a focused R&D effort that simultaneously measurably impacts the quality of life across society, demonstrates leadership and vision in large scale computing for the biological sciences, and trains the next generation of engineers, scientists, and mathematicians. To realize our goals we need to formalize the notion of “petascale computing” into an activity worthy of computer scientists’ and biologists’ dedicated effort, and that, in turn, enables them to succeed in their research career and in their core mission of training students. To create and strengthen a community, computer scientists and biologists, and their postdocs and students, should have the possibility of receiving travel and living support to visit each other or the petascale site for months at a time.

Long term funding, with a minimum of 5 years, will be needed to ensure continuity, and it should cover all participants: computer scientists, biologists, programmers, and so on.

This funding should encompass not only the research itself, but the sustained development of application and software libraries as they grow and require maintenance.

Projects should specifically be constructed as interdisciplinary teams working together to deploy biology applications at petascale. It will be crucial to have models for performance analysis and scaling to show that an algorithm scales *before* implementing it. There is also need to develop models for quantifying uncertainty (accuracy) and effort put into quantifying performance and accuracy of heuristics (performance modeling); these activities perforce require interdisciplinary collaboration.

## **2. Selection criteria for determining suitable petascale applications**

It is important that selection criteria lead to the formation of outstanding and productive collaborative teams focused on petascale activities. Historically, each 10-fold increase in parallelism required significant time investment, even when the raw compute capability was already available. Finally, since the petascale system will be a scarce resource during the first few years, its use must be managed transparently so it is in line with the selection criteria, and the needs of the funded efforts.

Therefore it is important to understand which applications are “peta ready” today (as was partially addressed by the list of applications compiled by Working Group I), and which will be ready tomorrow. Moreover, one must also prepare for the emergence of novel classes of calculations and applications that may be as important, or more, and as suitable for petascale as those of today.

One approach would be to distinguish among focused and large-scale development efforts and then apply the appropriate selection criteria and level of funding to each. For applications deemed “peta-scalable” by first-light (2011), the following criteria are important:

- a. qualifications of the team, size of the potential user base,
- b. suitability of the calculation for petascale resources,
- c. the balance between scientific merit and impact versus risk and feasibility.

Specifically, some of the most peta-scalable applications may be relatively low risk but lead to less practical results while other applications likely to be of great biological significance are clearly at the edge of feasibility. It is important that applications be evaluated along both dimensions of the risk-reward space and that the resulting portfolio mirror this spread.

The mix of applications may thus include some that are nearly ready for the petascale, others of high importance that require additional software engineering and algorithm improvement, or yet others that require a build-up phase towards substantially more compute-intensive usage. Other types of diversity to consider for support should include both monolithic single applications that spread across the entire machine, and heterogeneous applications where different components of the applications workflow stress the computer architecture in different ways.

## **3. Software challenges on the way to petascale**

An overwhelming consensus from the group, and indeed the workshop attendees across all groups, is that there is a need to increase support for theory, algorithm and code development, and software engineering to get the best science out of petascale and other high end machines.

A possible format for such support would be a two stage competition for software engineering centers, with the first stage being planning grants and the second stage being full software engineering centers, with each center focusing on a particular application or class of applications. The subject of these centers should include applications services as well as applications themselves, as both levels need considerable effort. There should be provisions for individual investigators who have distinct contributions to make, to contribute to the work of the centers without being an integral part of the centers.

Issues of intellectual property for software rights will be critical to work out in advance, as intellectual property considerations could cripple the dissemination of associated advancements in software, and the development of heterogeneous computing environments. One model for software intellectual property principles is embodied in the provisions for NIH software development grants.

To ensure that the biological sciences community is positioned to take advantage of petascale computing capabilities, it is important to support sustained efforts for the long term, i.e. a “20 years” timeline. A minimum of 5 years is recommended to establish retooling of (large) software, in order to successfully take advantage of what a petascale system has to offer. Other software will be enabled more quickly, but to focus on only these would be irresponsible to the field of biology and the opportunities being presented by establishing a petascale facility.

For support of applications, there is a need to invest in application libraries. With the help of source-to-source transformation (for example, Dan Quinlan's ROSE effort at LLNL) it is possible to implement semantic optimization of class libraries. Efforts of this type can greatly facilitate biologists, enabling them to work in terms of abstractions they are comfortable with. Additionally, however, biologists can greatly benefit from learning key aspects of computer science in focused training sessions key to their application areas (see below.)

Some problems pose challenges for hardware/algorithms (PME/Fast Fourier Transform) and may be suitable for more innovative algorithmic approaches/hardware. Several candidate applications do not fit into traditional numerical frameworks. For example, they may not be floating-point operation intensive or perhaps have poor spatio-temporal locality. This underlines the importance of identifying the computational needs of the candidate applications in advance of deployment to insure petascale infrastructure is appropriate to their needs (which should be among the first activities of collaborative groups). This issue is further addressed by the report of Working Group III below.

Many of the applications identified by Working Group I require multiscale modeling, relevant to many areas of science and engineering including but not limited to

computational biology (for example, the DoE has such a program for material science). In the case of computational biology, a grand challenge is to create multi-component, multi-dimensional, models spanning scales ranging from molecules to whole organ systems. There are various software and algorithmic challenges inherent in such models, in addition to the obvious physics/chemistry challenges of coupling across scale. In terms of the former challenges, for example, managing the components in a petascale system, in particular, “cross-component optimization”, is an important issue. That is, how to optimize multiple components taken together, rather than the classic approach of optimizing individual components separately. Thus, issues of load balancing, tolerating communication delays, memory hierarchy optimizations, etc., all increase complexity of the problem. All of these issues raise interesting research questions, as well as the need for software techniques needed to handle the optimizations. These issues are related to work being performed by the common component architecture (CCA) community, but complimentary in this domain. Frameworks are needed to express such optimizations, and to enable application developers to express, in the form of “application performance meta data”, the information needed to sensibly and collectively optimize the multiple components handling the different parts of a composed multi-scale simulation.

Scalability is a primary requirement for petascale; achieving it involves communications optimization, load balancing, asymptotic complexity analysis, and numerical accuracy assessment. When different models are composed, it is important that their numerical interactions be consistent, stable, etc. Models can not always be composed in a straightforward way, and it is important to understand their collective behavior and associated physical laws. Load balancing across different model; scales and data layouts is particularly hard and ill-understood at present. More research is needed in this domain, and this need is urgent in light of the number of likely processors in a petascale system (conservatively lower-bounded at 100,000).

Many of the issues arising out of multi-scale models are potentially relevant to *any* application running on a multi-scale system. Tolerating communication delays, and handling load balancing are much more difficult at extreme levels of parallelism ( $10^5$  to  $10^6$  processors), because done incorrectly, serious waste in compute resources results. Thus, software techniques that facilitate latency tolerance will play an important role in helping ensuring scalability and efficiency of resources.

#### **4. People and Infrastructure organization issues on the path to petascale**

Strategies for long term preparation of potentially new petascale applications include the development of training programs, workshops, and summer schools, with focus on teaching the craft of creating efficient codes. In parallel, sufficient effort needs to be devoted to the development of biological application middleware including appropriate run time systems, frameworks, and libraries. Leveraging of existing and successful programs in these areas is highly desirable.

There is an important need to pay careful attention to the development of a workforce pipeline for the applications software and engineering that are required. Proposals to existing programs, such as IGERT (Integrative Graduate Education and Research

Traineeship) are appropriate. In addition, more focused training programs that solicit proposals from multidisciplinary teams, to jointly train students in high performance computing for biology, should be considered. We have a consensus that high performance computing is relatively neglected in computer science departments around the country, and that this is a problem that should be addressed by incentives to train students in high performance computing.

There are also societal issues to be considered. Petascale computing is not simply an extrapolation of prior experience, but raises the stakes for application developers. Raising the scale of parallelism by a factor of 10 will compel some to rethink the algorithm, the implementation, or other software issues, including development costs. The impact of raising the level of parallelism by multiple orders of magnitude opens up considerable opportunity for the biology community to take advantage of. Taken from a different perspective, there are quite possibly unknown applications that become possible with the advent of petascale parallelism, which has the potential for opening up entirely new avenues for inquiry in Biology, Computer Science, Math, and Physics. These observations bring up two (possibly opposing) viewpoints.

- i. Petascale computing will start out as a small club gradually becoming more widespread over time, at which point we begin again with exaflop ( $10^{18}$ ) computing. It is important that a small number of likely success stories be chosen at the outset, endowed with the human resources needed to thrive, enabling the demonstration of the capabilities of the machine early on.
- ii. Appropriate training should be provided with a long term vision of developing new communities of users: to ensure that a critical mass of experts be available to respond to and disseminate information about new developments, and to develop new software techniques that will be useful to the computational biology community at large. Summer camps and summer schools should be held to teach computational biologists the latest software techniques, and to teach computer scientists the latest trends in computational biology. This should be an ongoing process. Resources should be set aside for innovative ideas to blossom, be they in computer science, computational biology, applied mathematics and physics, or other related fields. Small time users should be given the opportunity to use the full scale machine for trying out their ideas.

We believe there is a middle way, embodied in the proposal above to select a limited but diverse portfolio of applications and teams, and that this can foster both viewpoints. The result should be some early success at “first light”, but also more high risk, high reward research and innovation that can lead to new uses of the petascale computational facility to address emerging questions in biology.

## **II.C. Petascale Infrastructure**

This group dealt with identifying the computing, software, storage, networking, and people infrastructure needed for biology at the petascale level. A high-level take home message is *“invest in people and software at (or preferably) beyond the level of the hardware investment!”*

### **Hardware infrastructure**

Biological problems of interest, for example those identified by Working Group I, are data intensive, compute intensive, communication intensive, in variant combinations, and one size does not fit all. Biocomputing will therefore need multiple types of architectures and resources that map to the diverse hardware portfolio planned by NSF in their “tiers 1, 2 and 3” planning. However a form of “market segmentation” needs to be done to determine which calculations should be done where and to influence some of these architectures to be designed with the special needs of biocomputing in mind. This is related to the idea from Working Group II that interdisciplinary teams need to start by understanding, modeling, and extrapolating future application requirements before embarking on ambitious code development projects.

Several of the candidate applications described above by Working Group I will benefit from application-specific architectures (for example FPGAs can be programmed to do sequence comparisons very rapidly). Matching heterogeneity in applications and architectures across the NSF portfolio will be very important. It is expected that many candidate petascale applications in biology will be bandwidth intensive, with respect to local memory and with respect to inter-processor network bandwidth demands. In fact some key applications may turn out to be solved faster on what NSF terms “tier 2” systems than the petascale system, if those systems are better balanced in terms of memory and communications bandwidth per-processor. Therefore, it will be important to study computational characteristics of applications and associated hardware characteristics in advance to identify memory and communications bandwidth sensitivities. Likewise, it will be important to quantify what portions of candidate calculations are very computationally intensive and could be carried out on coprocessors (such FPGAs (Field Programmable Gate Arrays) and DSPs (Digital Signal Processors)), likely to become available in the same timeframe. Likewise, it will be important to understand which applications are very communications and I/O intensive and will stress machines in these dimensions. In the design of any petascale or “tier 2” I/O infrastructure, it will also be important to address data federation, data availability, and integrity. Also integration of data acquisition systems (sequencers, microarrays, imaging) needs to be addressed more than at present.

Deeper considerations of the specific needs of compute intensive versus data intensive biology applications need to be made, as these may not be easily separated. Data intensive applications require stable and scalable file systems, and infrastructure for moving the data in and out of the computer. When such an application is generating hundreds (or thousands) of petabytes, the infrastructure must support storing data, mining and analyzing data, moving data, archiving data, and visualizing data.

By the same token, compute intensive calculations come in different types requiring (1) considerable amounts data, (2) considerable numbers of CPUs, (3) considerable amounts of memory (4) real time/wall clock constraints and (5) combinations of all the previous. Also, even compute intensive applications are not always computing “just one number” as the output. Rather many will generate petabytes or more of output data even if they did not consume a similar amount of input data to start with. Thus even these may require data intensive post processing even if the petaflop calculation is not by itself data intensive.

A related issue, particularly in the cases of applications requiring the movement of vast amounts of data, concerns biology network issues. Schemes need to be developed for petabyte data transfer via the internet. This may require upgrades to existing national backbone networks but also, quite seriously, this may involve Fedex ala NetFlix (order data via the Web for next day arrival).

All of these challenges imply rethinking out-of-the-box around new architectures for biology computing, not just focus on refitting of existing biological problems to fit the petascale (or other high level) facility.

### **Software Infrastructure**

Software costs more than hardware. A strong consensus of the workshop participants is that currently there is an imbalance in NSF support for scalable, robust, easy to use scientific software relative to proposed investments in hardware. Enabling petascale computing in biology will require software infrastructure enabling data analysis, mining and visualization. Analyzing massive output data and visualizing will then require more than just high floating-point capability by way of infrastructure; candidate petascale biology applications present tremendous issues associated with data handling (federation of data sources as for example expression, sequence, phenotype, etc.). These applications will stress I/O and file systems, and data federation solutions. Algorithms will need to be developed to deal with uncertainty in data, missing data, and erroneous data (sensitivity analysis). Furthermore, there are two key issues involving interactivity around large-scale data: (1) inordinate amounts of data to move, store, analyze requiring infrastructure supporting interaction (2) human beings often will need to be in the analysis loop. Additional challenges are associated with the connection of sensors and data streaming, as data access rates for I/O become very important.

Associated networking software may not work without modification – For example TCP/IP has a 16 bit checksum and is subject to undetectable errors beyond 500 Gb packet transfers. ). Some applications will additionally bring real time constraints (epidemiology, customized care) and there will be a tradeoff between the model resolution as limited by these real-time constraints.

Viable infrastructure will also need to include scalable codes, scalable algorithms, and scalable memory as well as lots of cpus as was expounded upon by Working Group II above.

In addition to the significant work required in fundamental algorithms and load-balancing, latency tolerance methods, etc., as described by Working Group II, significant efforts need to be focused in the areas of queuing and scheduling. Currently, queuing and scheduling systems do not do a good job of handling different types of needs. There is minimal ability to schedule high-performance computers to accommodate real-time constraints, respond to embedded sensors, be available on demand, and the like uses of interest to biologists. Current scheduling policies primarily service throughput jobs. While it may be that this is also deemed to be the best way to manage the petascale system, there is significant doubt. Likely, increased programmer productivity, increased breakthrough science, and better response to the end-user may result from a less heavily loaded resource; one that is reserved for fewer truly petascale calculations, including perhaps some with real-time constraints, rather than the currently heavily loaded, highly utilized, NSF systems.

Additional issues related to software involve fault tolerance issues, which are currently not being adequately addressed in designing software infrastructure for petascale. Given likely state-of-the-art reliability and hardware failure rate trends, it is anticipated that one processor out of one hundred thousand (or a million) will fail every minute on a petascale machine. Who and how does one deal with such failures/minute? By way of example using current semantics, an MPI global operation will block if even one processor fails to respond resulting in code hang-up. Applications need to be re-written to be fault-tolerant, something currently not even possible without updating current semantics of MPI to enable more tolerant of failures. Either vendors need to develop, or the community needs to develop (more likely the latter) fault tolerant APIs and associated semantics for global message-passing systems, in order to enable large parallel codes to be re-written in a fault-tolerant style.

Generally speaking in high-performance computing (not just in biology), there is a dearth of scalable, robust, easy to use, interactive, etc. software tools. Many tools that do exist for the purpose are “professor-ware”, so there are lots of tools but not always with the required reliability or associated documentation. NSF could take the lead in finding mechanisms to fund enduring and stable tool efforts and in requiring periodic peer-review of ongoing tools projects.

### **People Infrastructure**

People cost more than hardware. It is important there be a proportional investment in the people - faculty, staff, and students - who will support the necessary and vital efforts to obtain petascale computing levels. Additionally, true peer collaboration is hard and circumstances must be fostered to overcome discipline silos. Echoing findings in Working Group II, interdisciplinary teams are necessary, involving biologists and computer scientists, but also other key disciplines (e.g., math, physics, sociology, economics, etc). Interdisciplinary teams may take the form of 2x2 collaborations as well as “service shop software models”. It is crucial these teams obtain early access to software and hardware at the teraflop level and higher on the path to petascale as it becomes available in order to prototype algorithms and software up to petaflop level computing.

## WORKSHOP PARTICIPANTS:

Aluru	Srinivas	Iowa State
Alessi	Sheila	UC San Diego
Alter	Orly	University of Texas
Baden	Scott	UC San Diego
Bader	David	Georgia Tech
Baldrige	Kim	University of Zurich
Beach	James	University of Kansas
Caldwell	Denise	NSF, MPS/PHY
Catlett	Charlie	Argonne National Laboratory
Chtchelkanova	Almadena	NSF, CISE/CCF
Cormier	Guy	Virginia Tech
Cummings	Michael	University of Maryland
Darema	Frederica	NSF, CISE/CNS
Dewaraja	Yuni	University of Michigan
Farwell	Sherry	NSF, EHR/EPSCoR
Feng	Wu	Virginia Tech
Fossum	Barbara	Purdue University
Grama	Ananth	Purdue University
Greer	Christopher	NSF, BIO/DBI, BIO/EF, OD/OCI
Gribskov	Mike	Purdue University
Heller	Miriam	NSF, OD/OCI
Henkart	Maryanna	NSF, BIO/EF, BIO/MCB
Istrail	Sorin	Brown University
Ivanov	Ivaylo	UC San Diego
Izaquirre	Jesus	Notre Dame
Jacobs	Gwen	Montana State
Jakkobsen	Eric	NCSA
Johnson	Chris	University of Utah
Karniadakis	George	Brown University
Koskella	Rebecca	Arctic Region Supercomputing Center
Lichtarge	Olivier	Baylor College of Medicine
Lyster	Peter	NIH
Maltby	Jim	Cray Inc.
Martinez	Neo	UC Santa Barbara, National Center for Ecological Analysis and Synthesis (NCEAS)
McCartney	Peter	NSF, BIO/DBI, Program Director
Meacham	Stephen	NSF, GEO/ATM, OD/OCI
Misawa	Eduardo	NSF, ENG/CMS
Munoz	Jose	NSF, OD/OCI
Myers	Gene	HHMI/JFRC
Nash	Stephen	NSF, ENG/DMI
O'Connor	Sally	NSF, BIO/DBI

Pennington	Deana	University of New Mexico / LTER
Pinkston	Timothy	NSF, CISE/CCF
Rajamony	Ram	IBM Corp.
Repasky	Dick	Indiana University
Reed	Daniel A.	University of North Carolina
Regetz	Jim	UC Santa Barbara, National Center for Ecological Analysis and Synthesis (NCEAS)
Roskoski	Joann	NSF, BIO/OAD
Samatova	Nagiza F.	Oak Ridge National Laboratory
Scheraga	Harold A.	Cornell University
Schulten	Klaus	University of Illinois
Shukla	Kamal	NSF, BIO/MCB
Silverthorne	Jane	NSF, BIO/DBI
Skolnick	Jeffrey	Georgia Tech
Snavely	Allan	UC San Diego
Stiles	Joel R.	Pittsburgh Supercomputing Center
Strohmaier	Erich	Lawrence Berkeley Laboratory
Taufer	Michela	University of Texas, El Paso
Voth	Greg	University of Utah
Walker	Ross	UC San Diego
Zimmerman	Ann	University of Michigan
Zorn	Manfred	NSF, BIO/DBI, Program Director