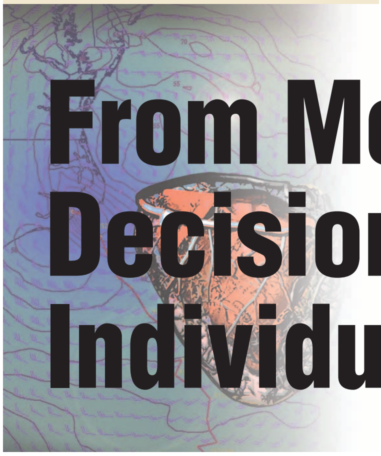
COVER FEATURE



From Molecule to Man: Decision Support in Individualized E-Health

Peter M.A. Sloot and Alfredo Tirado-Ramos, University of Amsterdam
Ilkay Altintas, University of California, San Diego and University of Amsterdam
Marian Bubak, AGH University of Science and Technology
Charles Boucher, Utrecht University

Computer science provides the language needed to study and understand complex multiscale, multiscience systems. ViroLab, a grid-based decision-support system, demonstrates how researchers can now study diseases from the DNA level all the way up to medical responses to treatment.

Complex human systems include unique and distinguishable components—from biological cells made of thousands of molecules, to immune systems built from billions of cells, to our society of more than 6 billion interacting individuals. Each gene in a cell, each cell in an immune system, and each individual in a society possesses characteristic behavior and provides unique contributions to the system.

The complete cascade—from genome, proteome, metabolome, and physiome to health—forms multiscale, multiscience systems and crosses many orders of magnitude in temporal and spatial scales,[1] as Figure 1 shows. The interactions between these systems create exquisite multitiered networks, with each component in nonlinear contact with many interaction partners. These networks aren't just complicated, they're complex. Understanding, quantifying, and handling this complexity is one of the biggest scientific challenges of our time.[2]

Computer science provides the language needed to study and understand these systems. Computer system architectures reflect the same laws and organizing principles used to build individualized biomedical systems, which can account for variations in physiology, treatment, and drug response.

PUSHING AND PULLING

An application pull has occurred in biomedicine with the move to in silico studies, which augment in vivo and in vitro studies by simulating more details of biomedical processes. Using these simulated processes helps medical doctors make decisions by exploring different scenarios. Preoperative simulation and visualization of vascular surgery[3] and expert systems for drug ranking[4] are two examples of such processes.

At the same time, a technology push is occurring in computing resources and data availability.[5] In the field of high-performance computing, as computing advanced from sequential to parallel to distributed, killer applications moved from mathematics to physics, chemistry, biology, and now to medicine. In addition, advances in Internet technology and grid computing[6] have made huge amounts of data available from sensors, experiments, and simulations.

Still, significant computational, integration, collaboration, and interaction gaps exist between the observed application pull and the technology push.

Bridging the gaps

Closing the computational gap in systems biology requires constructing, integrating, and managing a plethora of models. A bottom-up, data-driven approach

Published by the IEEE Computer Society

0018-9162/06/$20.00 © 2006 IEEE

won't work for this. Integrating often incompatible applications and tools for data acquisition, registration, storage, provenance, organization, analysis, and presentation requires using Web and grid services.

Even if we can solve the computational and integration challenges, we still need a system-level approach to close the collaboration and interaction gap. Such an approach would involve sharing processes, data, information, and knowledge across geographic and organizational boundaries within the context of distributed, multidisciplinary, and multiorganizational collaborative teams, or virtual organizations.

Finally, we need intuitive methods to dynamically streamline these processes depending on their availability, their reliability, and the specific interests of medical doctors, surgeons, clinical experts, researchers, and other end users. Scientific workflows, in which a workflow language expresses the flow of data and action from one step to another, provide one option for capturing such methods.[7,8] Figure 2 illustrates a general scheme for conducting e-science research.

ViroLab (www.virolab.org), a grid-based decision-support system (DSS) for infectious diseases, consists of modules, such as those that Figure 2 shows, for individualized drug ranking in human immunodeficiency virus (HIV) diseases. We used the complex HIV drug-resistance problem as a prototype for our system-level approach for two reasons. First, HIV drug resistance is becoming an increasing problem worldwide, with combination therapy with antiretroviral drugs failing to completely suppress the virus in a considerable number of HIV-infected patients. Second, HIV drug resistance is one of the few areas in medicine where genetic information is widely available and has been used for many years. As a consequence, large numbers of complex genetic sequences are available, in addition to clinical data.

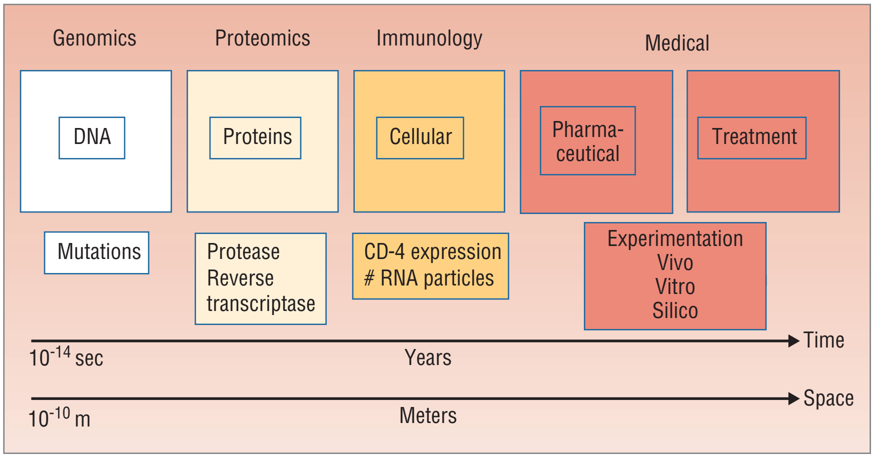


Figure 1. Time and space. Studying drug response in infectious diseases requires multiscale, multiscience models and techniques to cover the huge spatial and temporal scales.

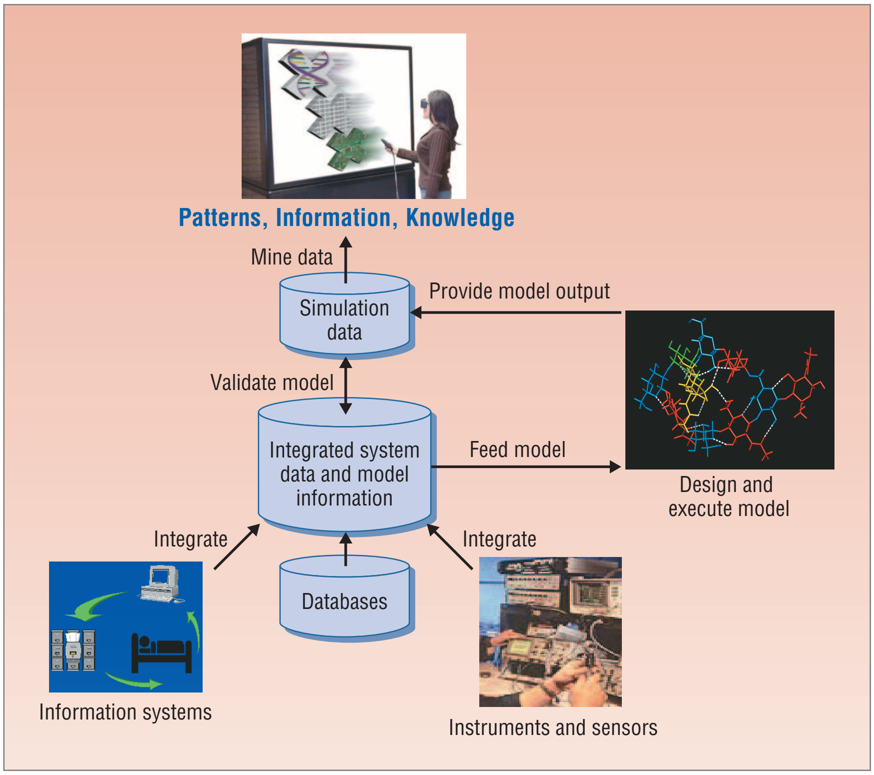


Figure 2. General architecture for e-science research. Information systems integrate available data with data from specialized instruments and sensors into distributed repositories. The systems then execute computational models using the integrated data, providing large quantities of model output data, which is mined and processed to extract useful knowledge.

COLLABORATIVE DSS

During the past decade, researchers have made significant progress in treating patients with viral diseases. For example, pharmaceutical companies now offer nearly 20 antiretroviral drugs for HIV treatment. However, to completely suppress the virus, patients must

take a combination of at least two of the four different classes of antiretroviral drugs.[9]

In a significant proportion of patients, however, the drugs fail to completely suppress the viral disease, resulting in the rapid selection of drug-resistant viruses and loss of drug effectiveness. This complicates the clinician's decision process, since clinical interpretation is based on data sets relating mutations to changes in drug sensitivity and relating mutations present in the virus to clinical responses to specific treatment regimens.

Interpretation tools

In recent years, researchers have developed several genotypic resistance-interpretation tools that help clinicians and virologists choose effective therapeutic alternatives. However, there's significant discordance among available systems for interpreting HIV genotypic resistance, for example. There's an urgent need for a joint effort to develop, validate, and publish standardized rules, as well as definition criteria for genotypic-resistance interpretation, and to provide accessible interpretation tools that help make genotypic assay results more clinically useful.

ViroLab offers clinicians a distributed virtual laboratory securely accessible from their hospitals and institutes throughout Europe.

Applying artificial intelligence and computational techniques to biomedicine has resulted in the development of computer-based DSSs. Recent developments in distributed computing further allow the virtualization of the massive data, computational, and software resources that complex DSSs require.

ViroLab's goal is to provide a virtual laboratory where researchers and medical doctors have easy access to distributed simulations and can share, process, and analyze virological, immunological, clinical, and experimental infectious disease data. Currently, virologists browse journals, select results, compile them for discussion, and derive rules for ranking and making decisions. ViroLab advances the state of the art by offering clinicians a distributed virtual laboratory securely accessible from their hospitals and institutes throughout Europe.

Under a typical usage scenario for ViroLab:

- A scientist from a clinical and epidemiological virology laboratory in Utrecht, Netherlands, securely accesses virus sequence, amino acid, or mutations data from a hospital AIDS lab in Rome using grid technology components running in Stuttgart, Germany.
- The scientist applies quality indicators needed for data-provenance tracking using provenance-server components running in Krakow, Poland.
- Researchers use this data as input to (molecular dynamics) simulations and immune system simulations running on grid nodes that reside at University College London and the University of Amsterdam.
- The virtualized DSS automatically derives metarules.
- Intelligent system components from Amsterdam use first-order logic to clean rules, identify conflicts and redundancy, and check logical consistency.
- The scientist validates new rules that the system automatically uploads into the virtualized DSS.
- The system presents a new ranking.

Advanced environment

ViroLab facilitates medical knowledge discovery and decision support for drug resistance, providing medical doctors with a rule-based, distributed DSS to rank drugs targeted at patients. Its infrastructure provides virologists with an advanced environment to study trends on an individual, population, and epidemiological level. That is, by virtualizing the hardware, compute infrastructure, and databases, the virtual laboratory will offer a user-friendly environment, with tailored workflow templates to harness and automate such diverse tasks as data archiving, integration, mining, and analysis; modeling and simulation; and integrating biomedical information from viruses (proteins and mutations), patients (viral load), and the literature (drug-resistance experiments).

A DSS and data analysis tools are at the center of the ViroLab distributed virtual laboratory. One such interpretation tool, Retrogram, estimates the sensitivity for available drugs by interpreting a patient's genotype using mutational algorithms that experts developed based on scientific literature, taking into account the published data relating genotype to phenotype. The ranking is also based on data from clinical studies of the relationship between the presence of particular mutations and the clinical or virological outcome.

For the system to support grid-based distributed data access and computation, virtualization of its components is important. ViroLab includes advanced tools for biostatistical analysis, visualization, modeling, and simulation that enable prediction of the temporal virological and immunological response of viruses with complex mutation patterns for drug therapy, as Figure 3 shows.

ViroLab architecture

In ViroLab, each experiment is a set of interconnected activities. The ViroLab system's design guarantees the interaction between a user and running applications, similar to methods used in real experiments, so the user can change a selected set of input data or parameters at runtime.

In addition to the DSS, patient databases, data analysis tools, and simulation software, ViroLab's runtime system consists of:

- a distributed, fault-tolerant registry for storing, updating, and publishing semantic information about available resources and executed applications;
- a tool to compose new experiments or modify experiments already performed;
- an execution engine to enact workflows according to data and action flow; and
- a scheduler for dynamic selection of resources for efficiently running a given experiment.

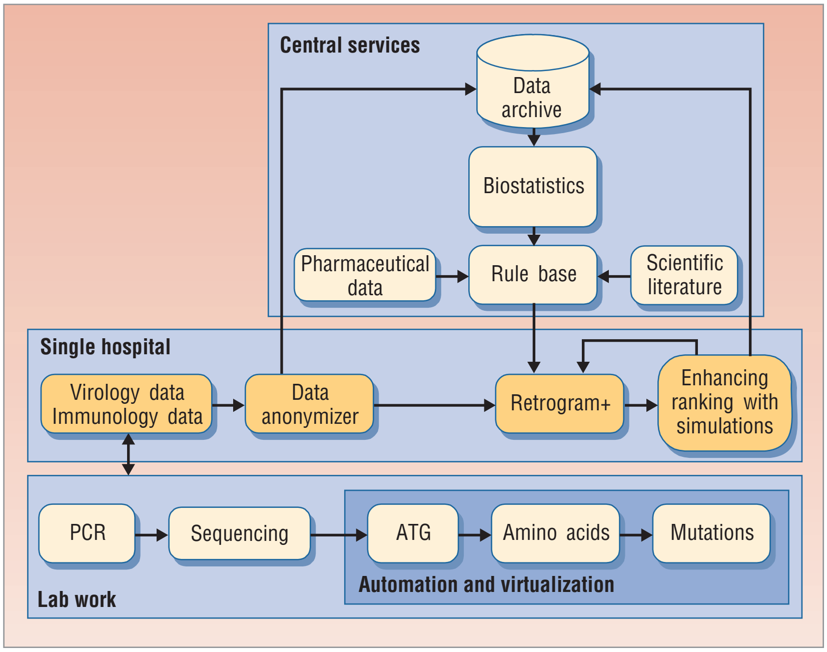


Figure 3. ViroLab data and control flow schematic. Manual wet lab is automated and virtualized, and the resulting data is fed to anonymizing components, as well as directly to the DSS to be ranked. Simulation components enhance output rankings, which are stored before being applied to rule-based algorithms and then fed back for prediction of the virus's drug sensitivity.

ViroLab workflows enable dynamic workflow execution, lazy scheduling, and runtime recomposition. They also support two levels of abstraction needed to operate separately on abstract workflows (workflow templates) and on concrete workflow instances (executables). Development of a virtual laboratory faces some major challenges, however, including:

- the highly distributed and heterogeneous nature of virological, immunological, clinical, and experimental data;
- the high dimensionality and complexity of the genetic and patient data; and
- the inaccessibility and (lack of) interoperability of advanced modeling, simulation, and analyses tools.

Recent advances in grid computing tackle these problems by virtualizing the resources (data, instruments, compute nodes, tools, and users) and making them transparently available. In grid computing, the virtual organization is the basic unit. Such an organization is a set of grid entities—individuals, institutions, applications, services, or resources—that are related to each other by some level of trust. Figure 4 summarizes these ideas.

ViroLab users can verify and identify the data's origin and rerun experiments when required. ViroLab extends this feature by categorizing the level of information, including the data and workflow process.

The collected data-provenance information is archived in ViroLab's portal and accessible through search and discovery methods. Examples of provenance information are:

- keeping track of the level of information to be saved,
- the format of information and where to save it,
- dynamic data and parameter changes during runtime and in time,
- saving workflow instances, and
- the information on how and by whom the run was made.

Technical requirements for building such a system include:

- efficient data management;
- integration and analysis;
- error detection;
- recovery from failures;
- logging information for each workflow;
- allowing status checks on running workflows;
- on-the-fly updates;
- detached execution of data- and compute-intensive tasks;
- visualization and image processing on the data flowing through the analysis steps;
- semantics; and
- metadata-based data access, authentication, and authorization.

Introducing different heterogeneous distributed network computing systems, data sources, and instruments creates additional technical challenges.

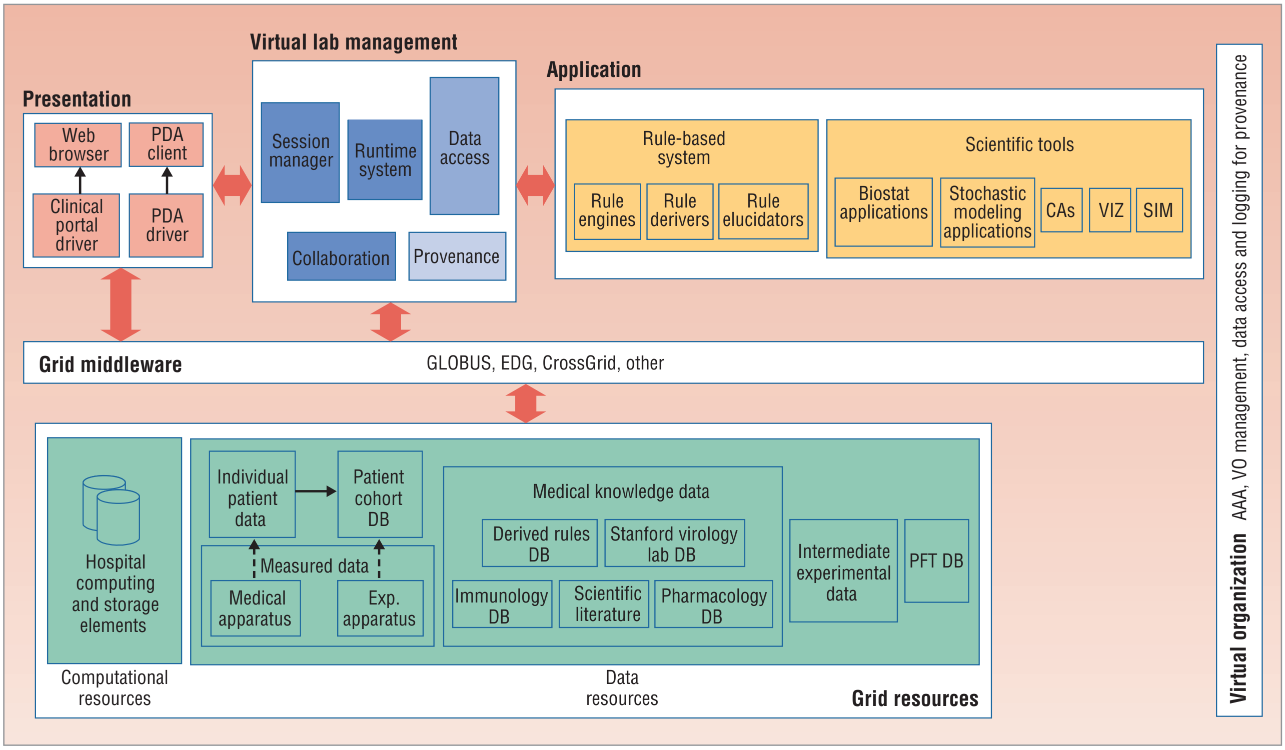


Figure 4. ViroLab system architecture. Distributed resources (computing elements, data, and storage) that the biomedical applications use are coordinated with the grid middleware and a grid runtime system.

ViroLab interactivity

In the ViroLab context, the availability of grid services and tools for interactive compute- and data-intensive applications presents an important research problem. Here we build on the European Union IST CrossGrid Project,[10] which developed a unified approach for running interactive distributed applications on the grid by providing solutions to the following issues:

- automatic porting of applications to grid environments;
- user interaction services for interactive startup of applications, online output control, parameter study, and runtime steering;
- advanced user interfaces that enable easy plug-in of applications and tools, like interactive performance analysis combined with online monitoring;
- scheduling of distributed interactive applications;
- benchmarking and performance prediction; and
- optimization of data access to different storage systems.

We recently tested these functionalities in a system that supports grid-based vascular reconstruction through bypass surgery by automating the process flow of MRI scan data, 3D visualization, and bypass creation and evaluation.[3] The developed computational components were executed efficiently as a custom-built application using the CrossGrid infrastructure, thus helping scientists carry out their scientific processing flows and run their analyses on both local and distributed resources. Virtual organization members with access to the resources the tasks were distributed to can reuse, share, and modify a process flow once it has been developed.

Workflows as system science language

An increasing number of computational tools for distributed computing in science have become available in recent years. However, they're mostly at an infrastructural level, making it difficult for the domain scientist to use them. Scientific workflow environments[11,12] improve this situation by allowing scientists to use different tools and technologies in a user-friendly, visual programming environment. These environments provide domain-independent, customizable GUIs for combining different e-science technologies along with efficient methods for using them, thus increasing efficiency and promoting scientific discovery.

A custom-built approach isn't sufficient for increasingly complex applications. Service-based distributed applications are ideal for automating and generalizing scientific workflows. Researchers can use them to combine data integration, analysis, and visualization steps into larger, automated "knowledge discovery pipelines" and "grid workflows."

One goal in building ViroLab's interactive scientific workflow environment was to add flexibility and extensibility, providing service-oriented interfaces through a workbench-style collaborative portal so that those with the right privileges can use the set of applications and data sets. An important issue is for users to be able to register and publish derived data and processes and to keep track of the provenance of information flowing through the generated pipelines, as well as accessing existing (patient and scientific literature) data and acquiring new data from scientific instruments. These domain-independent features can then be customized by adding domain-specific components and semantic annotation of the components and data being used.

Directly applying well-known mathematical approaches to analyze the HIV-1 genotype results in many problems.

Semantic assistance

To automate the construction of workflow applications, the system needs to generate ontological descriptions of services, system components, and their infrastructure. The OntoGrid project (www.ontogrid.net) and the Knowledge-Based Workflow System for Grid Applications (www.kwfgrid.net) both demonstrate these abilities. Semantic data usually is stored as a registry that contains Web Ontology Language (OWL) descriptions of service class functionality, instance properties, and performance records. The user provides a set of initial requirements about the workflow use, then the system builds an abstract workflow using the knowledge about services' functionality that service providers have supplied to the registry.

Subsequently, the system must apply semantic information on service properties, which results from analyzing the monitoring data of services and resources, to steer running workflows that still have multiple possibilities of concrete Web service operations. The system can select the preferable service class by comparing semantic descriptions of the available services classes and matching the classes' features to the actual requirements.

PRELIMINARY RESULTS

ViroLab uses statistical and immunological models to study the dynamics of the HIV populations and molecular dynamics models to study drug affinities, in addition to rule-based and parameter-based decision support. To enhance the analysis of highly dimensional, complex data, we added cellular automata and molecular dynamics modeling of HIV infection and AIDS onset to ViroLab.

HIV simulation

ViroLab uses a mesoscopic model to study the HIV infection's evolution and the onset of AIDS. The model takes into account the global features of the immune response to any pathogen, HIV's fast mutation rate, and a fair amount of spatial localization, which can occur in the lymph nodes. Ordinary (or partial) differential equation models can't sufficiently describe the two extreme timescales involved in HIV infection (days and decades) or the implicit spatial heterogeneity.

To study the dynamics of drug therapy for HIV infection, we developed a nonuniform cellular automata model that simulates four phases: acute, chronic, drug treatment response, and AIDS onset. Researchers also can use this model to study three different drug therapies: monotherapy, combined drug therapy, and highly active antiretroviral therapy. Our model for predicting the immune system's temporal behavior to drug therapy qualitatively corresponds to clinical data.[13]

Biostatistics

The biostatistical analysis of the HIV-1 genotype data sets aims to identify patterns of mutations (or naturally occurring polymorphisms) associated with resistance to antiviral drugs and to predict the degree of in vitro or in vivo sensitivity to available drugs from an HIV-1 genetic sequence. Analyzing this highly dimensional data presents a statistical challenge.[14]

Directly applying well-known mathematical approaches to analyze the HIV-1 genotype results in many problems stemming from the fact that in HIV DNA analysis, relevant mutations—a set of mutations associated specifically with the drug resistance—are the main scope of interest. These mutations might exist in different positions over the amino-acid chains. Moreover, the sheer complexity of the disease and data require development of a reliable statistical technique for its analysis and modeling.[4]

DSS and presentation

The output of our initial ViroLab version consists of a prediction of the virus's drug sensitivity generated by comparing the viral genotype to a relational database containing a large number of phenotype-genotype pairs. The decision software interprets a patient's genotype by using rules developed by experts on the basis of the literature, taking into account the relationship of the genotype and phenotype. In addition, the output is based on available data from clinical studies and on the relationship between the presence of genotype and the clinical outcome.

Researchers can use a Proxy and Java 2 Micro Edition method to access ViroLab from mobile devices, thus lowering system access barriers. A mininavigator script communicates patient data with the remote server, where the ranking takes place.

With the increasing availability of genetic information and extensive patient records, researchers can now study diseases from the DNA level all the way up to medical responses. Resolving the long-standing challenges of individual-based, targeted treatments is coming within reach. It's necessary to provide integrating technology to the medical doctors and researchers bridging the gaps in multiscale models, data fusion, and cross-disciplinary collaboration.

Although the ViroLab research is still in its infancy, results indicate that our personalized drug ranking prototype is viable and extensible. The system remains under development, with new functionalities being added from usability studies in a network of European hospitals. ■

Acknowledgments

The authors thank the ViroLab consortium, in particular, Dr. D. van de Vijver from University Medical Center Utrecht. This research was supported by the Dutch Virtual Laboratory for e-Science project and the European Union ViroLab grant INFSO-IST-027446. We thank Carl Kesselman and Ian Foster for proofreading and offering many valuable suggestions to improve the overall quality of the article.

References

1. A. Finkelstein et al., "Computational Challenges of System Biology," *Computer*, May 2004, pp. 26-33.
2. A-L. Barabási, "Taming Complexity," *Nature Physics*, Nov. 2005, pp. 68-70.
3. A. Tirado-Ramos et al., "An Integrative Approach to High-Performance Biomedical Problem Solving Environments on the Grid," *Parallel Computing*, Sept./Oct. 2004, pp. 1037-1055.
4. P.M.A. Sloot et al., "A Grid-Based HIV Expert System," *J. Clinical Monitoring and Computing*, Oct. 2005 pp. 263-278.
5. A.J.G. Hey and A.E. Trefethen, "The Data Deluge: An e-Science Perspective," F. Berman, G.C. Fox, and A.J.G. Hey, eds., *Grid Computing–Making the Global Infrastructure a Reality*, Wiley & Sons, 2003, pp. 809-824.
6. I. Foster, C. Kesselman, and S. Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations," *Int'l J. Supercomputer Applications*, fall 2001, pp. 200-222; http://globus.org/alliance/publications/papers/anatomy.pdf.
7. L. Altintas et al., "A Framework for the Design and Reuse of Grid Workflows," *Proc. Scientific Applications of Grid Computing*, (SAG 04), LNCS 3458, Springer, 2005, pp. 119-132.
8. F. Neubauer, A. Hoheisel, and J. Geiler, "Workflow-Based Grid Applications," *Future Generation Computer Systems*, Jan. 2006, pp. 6-15.
9. S.G. Deeks, "Treatment of Antiretroviral-Drug-Resistant HIV-1 Infection," *Lancet*, 13 Dec., 2003, pp. 2002-2011.
10. M. Bubak, M. Malawski, and K. Zajac, "Architecture of the Grid for Interactive Applications," *Proc. Int'l Conf. Computational Science* LNCS 2657, Springer, 2003, pp. 207-213; www.crossGrid.org.
11. B. Ludäscher et al., "Scientific Workflow Management and the Kepler System," *Concurrency and Computation: Practice & Experience*, Wiley & Sons, 2006, pp. 1039-1065.
12. M. Bubak et al., "Workflow Composer and Service Registry for Grid Applications," *Future Generation Computer Systems*, Jan. 2005, pp. 79-86.
13. P.M.A. Sloot, F. Chen, and C.A. Boucher, "Cellular Automata Model of Drug Therapy for HIV Infection," S. Bandini, B. Chopard, and M. Tomassini, eds., *Proc. 5th Int'l Conf. Cellular Automata for Research and Industry* (ACRI 02), Springer, LNCS 2493, Springer, 2002, pp. 282-293.
14. T.E. Scheetz et al., "Gene Transcript Clustering: A Comparison of Parallel Approaches," *Future Generation Computer Systems*, May 2005, pp. 731-735.

***Peter M.A. Sloot** is a computational sciences professor at the University of Amsterdam's Informatics Institute. He received a PhD in computer science from the University of Amsterdam. Sloot is a member of the IEEE. Contact him at sloot@science.uva.nl.*

***Alfredo Tirado-Ramos** is a PhD candidate at the University of Amsterdam. Tirado-Ramos is a member of the IEEE and the American Medical Informatics Association. Contact him at alfredo@science.uva.nl.*

***Ilkay Altintas** leads the San Diego Supercomputer Center's Scientific Workflow Automation Technologies Lab at the University of California, San Diego, and is assistant director of the National Laboratory for Advanced Data Research. Altintas is an external PhD student in computational sciences at the University of Amsterdam's Informatics Institute. She is a member of the IEEE and the ACM. Contact her at altintas@sdsc.edu.*

***Marian Bubak** is an adjunct professor at the Institute of Computer Science and a staff member of the Academic Computer Centre Cyfronet at the AGH University of Science and Technology. He received a PhD in computer science from AGH. He is a member of the CoreGRID Integration Monitoring Committee. Contact him at bubak@agh.edu.pl.*

***Charles Boucher** is an associate professor in the Department of Medical Microbiology at Utrecht University. Boucher received an MD and a PhD from the University of Amsterdam. He is a member of the National Institutes of Health, the International Virology AIDS Clinical Trials Groups, and the International AIDS Society. Contact him at C.Boucher@azu.nl.*