

Position Paper
Cyber-Infrastructure for Pervasive Computing: Opportunities and Challenges
Gerhard Klimeck, gekco@purdue.edu

Premises:

Infrastructure as defined in the Merriam-Webster online dictionary: “the system of public works of a country, state, or region; also: the resources (as personnel, buildings, or equipment) required for an activity.”

Today’s infrastructure of desktop computing for publications and presentations is almost as pervasive as the infrastructure of roads in the United States. Similarly pervasive is the use of advanced gaming engines in households with children, where advanced computer systems offer amazing networked virtual realities for entertainment and education. However, that is not true for the use of scientific computing to obtain and disseminate knowledge in science and engineering or even of the general public. Scientific computing remains the domain of a relatively small number of experts and only very few engineering disciplines (fluid dynamics) have embraced the use of supercomputers. The state of “the grid” is such that only very specialized researchers can benefit from the grid compute power. Queues are full of large parallel jobs, while the masses cannot get their relatively simple calculations done. Getting calculations done rapidly within seconds or minutes, is a critical element in the interactive exploration of the design space for most real-world applications. Second stage analysis might happen over the lunch time or overnight runs. Extremely few people have the patience and stamina for computations that take days, weeks, or months. The true opportunity for HPC is to reduce the hour or day long runs into minutes, and to reduce the very long compute times to a lunch hour.

Script-based computational environments like Matlab, Mathematica, or python have broken a new ground for scientists who normally would have never touched a computer. However, some high level programming skills are needed to create new “workflows”. The true potential impact of computing on science and engineering will come when users are enabled to ask science or engineering questions to applications rather than having to build applications themselves.

I attribute the lack of broad impact of computation to three technical and one sociological factors: 1) the limited availability of A) compute cycles, B) reliable middleware, and C) user interfaces for researchers and educators who are not computational experts, and 2) a very limited reward mechanism for the research community to truly engage in outreach to make their fascinating research results available to others.

Cyber-Infrastructure Opportunities:

Most adults in the US can operate a very broad set of cars without reading a manual or specialty training for specific vehicle and make that infrastructure part of their daily business operation. Computational models and compute cycles need to be available to a broad set of researchers, educators, and students in the same sense of day-to-day operation. If we reach that point we have made computing pervasive and we will have created a new paradigm in research and education. Modeling and simulation will have augmented experiments and education as an integral part and will be part of our economic development. Data and models will widely available and *usable*. The automobile transportation system has lifted the US economy to new capabilities. Society is willing to pay the price, even accidents that occur. Similarly, pervasive computation can change the way we do science, engineering, and business in a very positive way. Of course, there will be a price; limited misuse of the computational systems will be part of life as well.

The next two pages lay out the challenges in more detail and suggest the actions NSF and the TeraGrid could take to overcome them, to make scientific computing truly pervasive.

1. Cyber-Infrastructure Challenges:

A hundred years ago it was accepted that people who want to drive an automobile must have a professional driver and be quite rich. Eventually the operation of automobiles was standardized and the use of such an infrastructure is now second nature to us. Analogously we are today limiting resources for computational science to a very specialized and highly trained group of individuals and the common consensus is that it needs to stay that way. The primary *technical* limiting factors in the creation of a truly broadly used infrastructure is the availability of compute cycles, availability of reliable middleware, and availability of applications that make cyber-infrastructure an integral part of daily life (e.g. Global Positioning System, Google, Spreadsheets, e-mail, the Web, etc.) – think of applications like using computers to keep people much more healthy, businesses or local economies better planned, designs more reliable, etc. The issue is the availability of software that can be used by a broad community.

1.A. Provision of Compute Cycles:

National computational resource providers remain primarily focused on serving a very small number of high-end, high-profile computational researchers. The proliferation of relatively small computational cluster resources is indicative of the failure of national compute centers to deliver services to the broad community. However, the typical cluster solutions impose a large burden on the research system, because they consume a lot of start-up and research grant funding, they consume graduate student investments (since they are typically not professionally administered), and a vast amount of compute cycles are wasted due to high idle times. But why are they so pervasive? For the broad community the key element of success is rapid turn-around of computation on transparently available compute resources. National resource providers must seize this opportunity to serve cycles in a manner that researchers prefer to run on centralized systems, rather than personal, expensive, badly maintained, and underutilized systems.

Virtualization of centralized compute resources can result in highly customizable, user-oriented systems that can be dynamically allocated and migrated. If the access looks and feels the same as having the machine in the basement, then users might be willing to come to the national compute resource providers for service.

The TeraGrid has certainly not embraced this opportunity although significant amounts of compute cycles are apparently not being allocated or used. Campus grids must be integrated into the national framework to carry computational loads on an exchange-like service agreement.

1.B. Connecting resources and users through middleware:

Middleware is the software that connects applications, compute cycles and user interfaces. The connection must occur in a transparent way - just like a cell phone call, which can reach any other telephone, landline, or cell in the world. Today's state of "the grid" is comparable to that of telephony in the 1920's where rich people had phones, calls were placed through an operator who had to be intimately involved, and quality and reliability were poor.

The existing grid computing concepts are purely focused on batch-style computing and seem to completely ignore the on-demand delivery of service, immediate feedback from the code, or even simple reporting of consumed compute cycles. The integration of graphical user interfaces as front end to computational engines is a critical means to enable interactive model and data exploration. The concept of community accounts is the critical means to open compute resources to a broad set of users like the nanoHUB. However security models appear to solely focused on the need of the resource provider and not on the need of applications to be deployed rapidly by a community. NSF has invested significant resources in middleware *research*, which resulted in a large set of publications yet very fragile middleware software that

is purely geared towards delivery of compute cycles to experts. What is missing is a serious effort on middleware development by software professionals with the goal of robust, interactive software. The emphasis on those projects must be on infrastructure not research.

Software is infrastructure! NSF should focus on funding on the development of robust production-level middleware. The TeraGrid should serve as a ground for exploration of future standards, rather than imposing fragile standards at this time.

1.C. Provision of end-to-end applications

The TeraGrid formally embraces the support of Science Gateways as access points for computational resources. Such gateways have the potential to open the TeraGrid to tens of thousands of users in the broader science and engineering community. However, the overall funding for the support of gateways is small and not of critical mass to really make a difference. In fact funding has been pulled away from currently successful gateways and centralized into a single gateway unit.

The nanoHUB is an example of what can be done if the deployment focus is on the usability of the tools for non-expert users. Over 5,800 users ran over 240,000 simulations in 2007 completely transparently in an end-to-end service, without installing any software on their client side computers. The key insights from the nanoHUB experience is that the software has to be more than available, it has to be user friendly and interactive.

TeraGrid should direct significant funding to science gateways to deliver computing to the broad scientific and engineering community, instead of a few experts. The development of graphical user interface standards is critical if the usage is to grow.

2. Changing the culture of computational researchers:

Research is driven by a person's desire to gain understanding or to develop new technologies. As we progress in the advancement of knowledge we look towards the challenges ahead of us, but not necessarily what impact our work could have if we were to put it into the hands of others. As such most software developed in a Ph.D. thesis is pretty much user hostile, limited to very few elite people, and lost with the graduation of the student. There must be a better reward mechanism for faculty members to convert more (but not necessarily all) the relevant research outcomes into true outreach.

Making results and software available is not only good for the broader research community; it should be deemed essential for the computational researchers, as this would make the science repeatable and transparent! This indeed would establish Computational Science as a third leg of science next to theory and experiment.

Academia is an industry that sells education and the advancement of science, engineering, and humanities. True outreach and economic development is typically not in the researchers reward system. The reward system is currently not laid out to truly promote the conversion of research results into use in related areas or even broader use by society.

Changes in the funding philosophy of major research agencies is needed to drive changes in that reward system of true outreach. The conversion of research insights, software, and data into content that others can use must be the key outcome of any research grant. Given such new boundary conditions, academia will indeed adjust!