Editors: Paul F. Dubois, paul@pfdubois.com
David Beazley, beazley@cs.uchicago.edu

DISTRIBUTED COMPUTING FOR PUBLIC-INTEREST CLIMATE MODELING RESEARCH

By Dave Stainforth, Jamie Kettleborough, Myles Allen, Mat Collins, Andy Heaps, and James Murphy

THE RAPID INCREASE IN THE SPEED AND CAPACITY OF COMMONLY AVAILABLE PCS IS PROVIDING AN OPPORTUNITY TO USE DISTRIBUTED COMPUTING TO TACKLE MAJOR MODELING TASKS SUCH AS CLIMATE SIMULATION. OUR CLIMATE*PREDICTION*.COM project is developing the software necessary to carry out such a project in the public domain.[1] In this article, we describe the development of the demonstration release software, along with the computational challenges such as data mining, visualization, and distributed database management that the project will address in the future.

The project

Historically, complex climate models (that is, combined atmosphere–ocean global circulation models) have run only on supercomputers. Recently, however, the increase in speed and memory of the typical home or business PC has opened up the possibility of undertaking serious modeling projects on such equipment. Projects such as SETI@home,[2] fightaidsathome, and Cure Cancer have demonstrated the principles and techniques of distributed computing, which provides a mechanism for researchers to tap into the unused resource of millions of home and business PCs. These projects tend to require small, frequent downloads of data that an analysis procedure on the participant's machine (or client) tests. The analysis then returns a result of the form "match" or "no match." Task units typically take a few hours to process, so there is no need for a long-term commitment from the participants, who are not themselves actively involved in the experiment.

The climate*prediction*.com project takes the distributed computing paradigm a step further by inviting participants to download a full-scale climate model and run it locally to simulate 100 years of the Earth's climate, from 1950 to 2050. Each participant carries out a unique simulation, which is then combined with other participants' simulations to form an *ensemble* climate forecast. Because each machine has the complete model, each participant's simulation is independent—there is little need for network data transfer during the run. This is in contrast to the parallelized version of such models, typically run on massively parallel processor (MPP) machines or Beowulf clusters, where high-speed processor connectivity is crucial.

The 100-year simulation takes roughly three months on a 1.4-GHz machine, using approximately 55 Mbytes of memory. Allowing for less than 100 percent CPU availability and for slower and faster machines, we estimate that participants will need to stay with the project for three to eight months. Those with slower machines might be allocated shorter runs within the experimental design.

It is a major and continuing task to bring together the various aspects of this project, from climate modeling under the Windows operating system, to server distribution and database management, to simulation visualization, to visualization and analysis of the whole ensemble. However, we are currently testing an initial version of the project, and the first public release is planned for later this year.

Experimental justification and design

The Inter-Governmental Panel on Climate Change (IPCC) published its Third Assessment Report in 2001,[3] which assessed the currently available scientific information on climate change. One of the areas it highlighted for priority action was the improvement of "methods to quantify uncertainties of climate projections and scenarios, including long-term ensemble simulations using complex models."[3] This is the main scientific goal of the climate*prediction*.com project.

Worldwide, only a handful of complex climate models exist, but over the last 10 years, they have developed to a state where they can simulate certain aspects of observed climate change over the last century with remarkable accuracy (see Figure 1). Having demonstrated their ability to simulate past changes, these models can be used to make projections of future climate changes under various scenarios for greenhouse gas emissions (see Figure 2).

1521-9615/02/$17.00 © 2002 IEEE

Café Dubois

Episode IV: A new hope

I am delighted that David Beazley is joining me as department editor. David is an assistant professor of computer science at the University of Chicago with significant experience in scientific programming. We will take turns writing this sidebar on an irregular schedule and share the other duties.

David and I met through a mutual interest in code-steering for scientific projects. David worked on a project at Los Alamos when he was a graduate student, and as a byproduct of his work produced a great tool, SWIG (Software Wrapper Interface Generator; swig.org), which lets you interface various interpreted languages such as Python and Perl to C libraries.

Among other things we've enjoyed doing together, I was a technical reviewer on both editions of his book, *Python Essential Reference* (New Riders Press, 2001), which is my constant companion. More recently he has contributed a section on XML to Steve Holden's new book, *Python Web Programming*.

David is a great writer and speaker. At the Python conference in March 2000, his talk actually elicited audible groans from that very skilled audience, who were simply amazed at how difficult a task he had accomplished. (It was a debugger that could retrace from Python down into C extensions and show the stack frames in *both* languages.) He got a standing ovation at the end and won the vote of the attendees for the Best Paper award.

I think you are going to enjoy reading more of his writing, and his academic position and different interests should balance out this department. I hope you read his great article on linking last year (vol. 3, no. 5, Sept./Oct., pp.90–97)—it was one of my all-time favorites. Please give David a warm welcome.

When America loved smart children

I grow old…I grow old…
I shall wear the bottoms of my trousers rolled.
—T.S. Eliot, "The Love Song of J. Alfred Prufrock," 1917

I was able to read Eliot in Mr. Tranchina's 10th grade English class because of Sputnik. After Sputnik went into orbit, Congress passed the National Defense Education Act. Every ninth grader in the country was given an IQ test, and those who scored high got to have some small classes with the best teachers and hard curricula.

For a tiny moment in time, America loved its smart children. In that window that Sputnik opened for me, I had physics at an observatory, including an extra hour a day building radios and the like; a chemistry class with dangerous chemicals; English with Dickens, Hugo, Dostoyevsky, and serious writing; and a mathematics class that did the normal stuff in six weeks and went on to series and groups and calculus at UC Berkeley. All of that in a public school in Oakland, California. How impossible that seems now.

It was glorious and crazy. I couldn't write like Cheryl, who wrote a novel that Mr. Tranchina said the rest of us could not read because it was too "adult;" my lab partner blew up our radio every time I turned my back on her, cutting short my physics career; and when I came in late and could not find any one-molar acid and zinc plate, I naively used 16-molar acid and zinc powder, ending my chemistry career and the lab ceiling.

Biology was out, too. The biology teacher thought that these IQ tests were accurate down to the last digit. He divided your test score by your IQ to "make it fair." He ate odd sea-creature things to make us sick. He noticed this girl and I had the exact same IQ scores, so he suggested I ask her out. Today is our 36th wedding anniversary. The teacher ended up in an asylum.

The other sciences ruled out, I said I wanted to be a mathematician. On Career Day, they sent me to the Frieden Calculator Company, where they told me that they didn't have any mathematicians and their one engineer was out that day. They couldn't think of any business that needed a mathematician. I decided I would teach high school.

Forty years later, I grow old. I often forget to unroll my trousers after riding my bike. I'm starting to plan for retirement, although that is some years away yet. (Part of that planning was making sure that the question, "What if you get hit by a truck?" now has a satisfactory answer: "David Beazley.") Thinking that maybe it was time to give back some of what I was given, I looked into high school teaching.

Wrong, sit down. I'm not qualified. It turns out that even if I took the education courses to compensate for the seven years of bad training I got teaching college, I am not qualified to teach your kid computer science. I never had 18 units of it. I'm 18 short. I guess I'll have to wait until some terrorist orbits a satellite.

Meantime, maybe I can write a better novel than Cheryl. How about that title that Tom Lehrer once suggested, *Tropic of Calculus*? While I do that, take this month's article to your local high school teacher and give those smart kids our love.

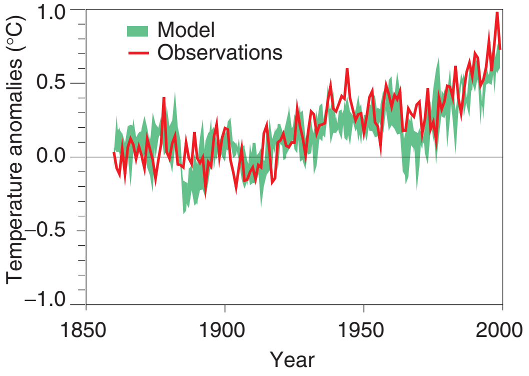

Figure 1. A simulation of the Earth's surface temperature variations from 1850 to 2000. The shading indicates the range of uncertainty from four simulations using the same complex model but with different initial conditions. Figure courtesy of the Inter-Governmental Panel on Climate Change.

There is, however, reason to believe that the models currently available are not the only complex climate models that we could devise consistent with the current observational record.[4] Other models could give quite different projections of future climate change. Thus, there are no objective uncertainty limits on the current projections. This is of great importance because although observations make it clear that anthropogenic greenhouse gas emissions affect the Earth's climate,[3] politicians and policy makers need estimates of uncertainty to help plan and justify action or inaction.

Devising a whole set of climate models is possible because such models have parameterized representations of many of the physical processes important in the climate system. These include cloud and precipitation processes, aspects of the hydrological cycle, and small-scale turbulent fluid mechanical processes in both the atmosphere and ocean. Many of the parameter values used are poorly determined, and there are several different ways of parameterizing the same process. By changing the values of parameters within their uncertainty limits or by changing the parameterizations themselves, we can create a large ensemble of different climate models. This is often referred to as a *perturbed physics ensemble*. There are objective methods available to compare the simulation results with observations and reject any unrealistic models. We can then analyze the remaining ensemble of models to assess whether it covers the range of behavior consistent with the observations. Subsequently, the realistic model versions can be used to simulate the future under various emission scenarios. The spread of these simulations gives an objective quantification of the uncertainty in the projections they make. In other words, we can use the combined ensemble to make a climate prediction.

Unfortunately, there are hundreds of uncertain parameters within such models and because they potentially interact nonlinearly, there are many millions of possible combinations. Furthermore, to accurately compare simulations with observations, we must repeat each perturbed physics simulation with several different sets of initial conditions. However, it could be that many parameters will not be independent, and intelligent sampling of parameter space should help reduce the number of runs necessary. Nevertheless, analysis of this system will still require 1 to 2 million independent simulations, which is far beyond the scope of the supercomputer facilities currently available.

We propose to approach the problem by a series of separate experiments

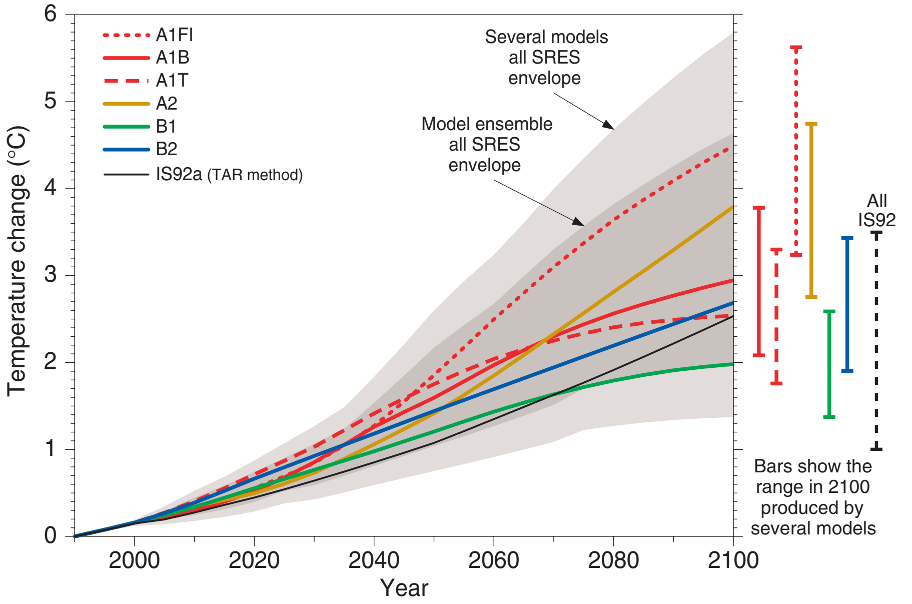

Figure 2. A comparison of projections of global mean surface temperature under various scenarios (labeled A1F1 to IS92a) for future levels of greenhouse gases, based on scenarios for policies and practices adopted globally. The shaded area is the current model uncertainty. These results are based on a simple climate model tuned to several complex models. Figure courtesy of the Inter-Governmental Panel on Climate Change.

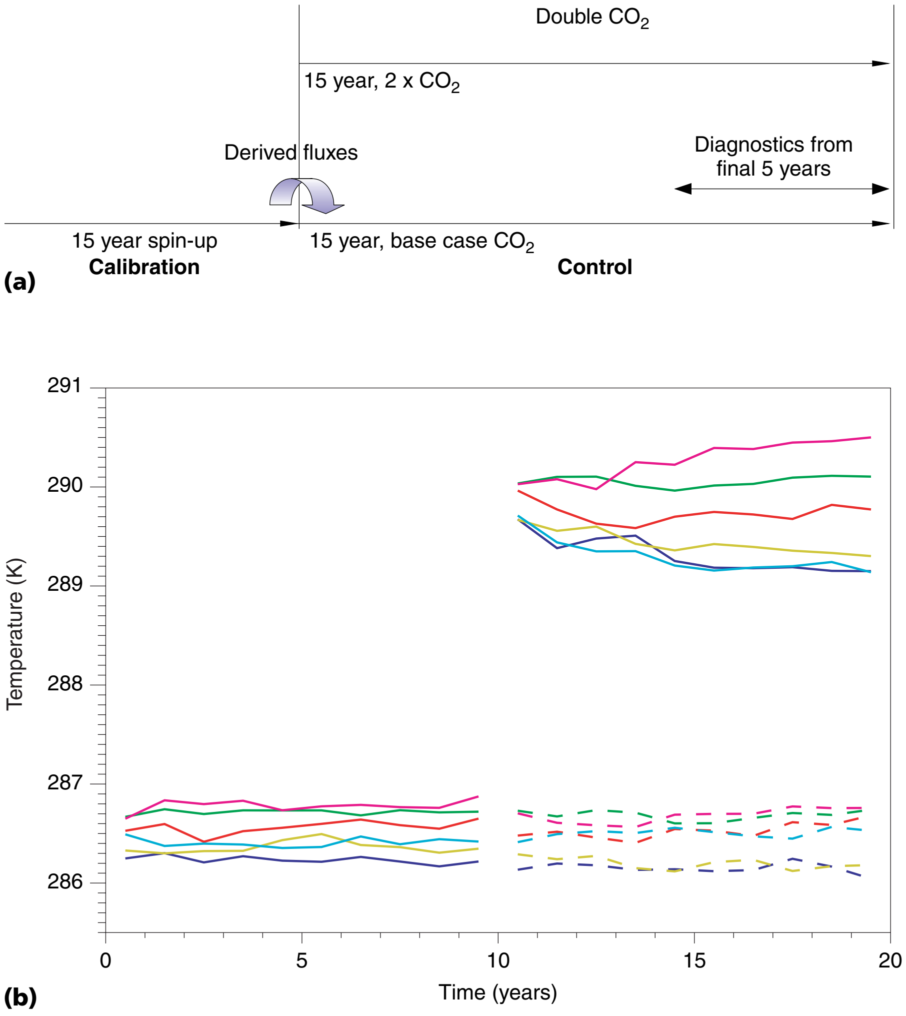

Figure 3. The slab model experiment. (a) The initial release consists of three phases. During phase 1, the sea-surface temperatures are relaxed to observed values. The atmosphere–ocean heat fluxes are derived from this phase and used to ensure a stable climate in phases 2 and 3, which have pre-industrial and doubled levels of CO_2, respectively. (b) The 1.5 m global annual mean surface temperature from six perturbed physics simulations of the slab model. Five-year means from the ends of phases 2 and 3 can be used to deduce the model's sensitivity to doubling CO_2.

using different versions of the Hadley Centre Climate Model, the global setup of the UK Met Office Unified Model.[5] This is one of the world's major complex climate models, developed over more than a decade by a large team of researchers. It consists of over 500,000 lines of Fortran and is designed to run on either a single processor or in MPP mode on supercomputers or clusters.

Demonstration and initial releases will use a version of the model where a climate resolution atmosphere (3.75° × 2.5°) is coupled to a single-layer thermodynamic "slab" ocean.[6] This setup has the benefit that it reaches a steady state quickly and can thus help evaluate climate sensitivity to doubling CO_2 levels with only 45 years of simulation. An initial 15-year run deduces the heat flux between the atmosphere and the ocean, which is necessary to keep the system stable and consistent with sea-surface temperature observations. This is followed by two 15-year runs with standard and doubled levels of CO_2 (see Figure 3). The results from these experiments will be valuable in themselves and will also be needed in the main experiment.

The main experiment consists of a 100-year simulation that uses a version of the model with the same atmosphere coupled to a full dynamic ocean. It will use two ocean resolutions: 1.25° × 1.25° and 3.75° × 2.5°.[7,8] The version with the higher resolution ocean runs at approximately half the speed and requires roughly twice the memory (110 Mbytes)—it is only suitable for relatively high-specification machines. The variation of ocean resolution can itself be treated as another parameter variation in the context of this work.

Participants will first run a 50-year simulation of the 1950 to 2000 period. The client software will automatically compare the simulation's results with observations of variables such as surface temperature, and the returned results will include a quality-of-fit weighting. Those versions with the closest fit to, and spanning the behavior constrained by, observations will be redistributed to simulate the 2000 to 2050 period. Each participant will have a unique version of the model thanks to the combination of perturbed physics, initial condition variations, and scenarios for past and future greenhouse gas levels (see Figure 4). The result will enable an objective analysis of uncertainty for climate projections to 2050 for a range of future emission scenarios. The data from such a large ensemble will also facilitate several other research

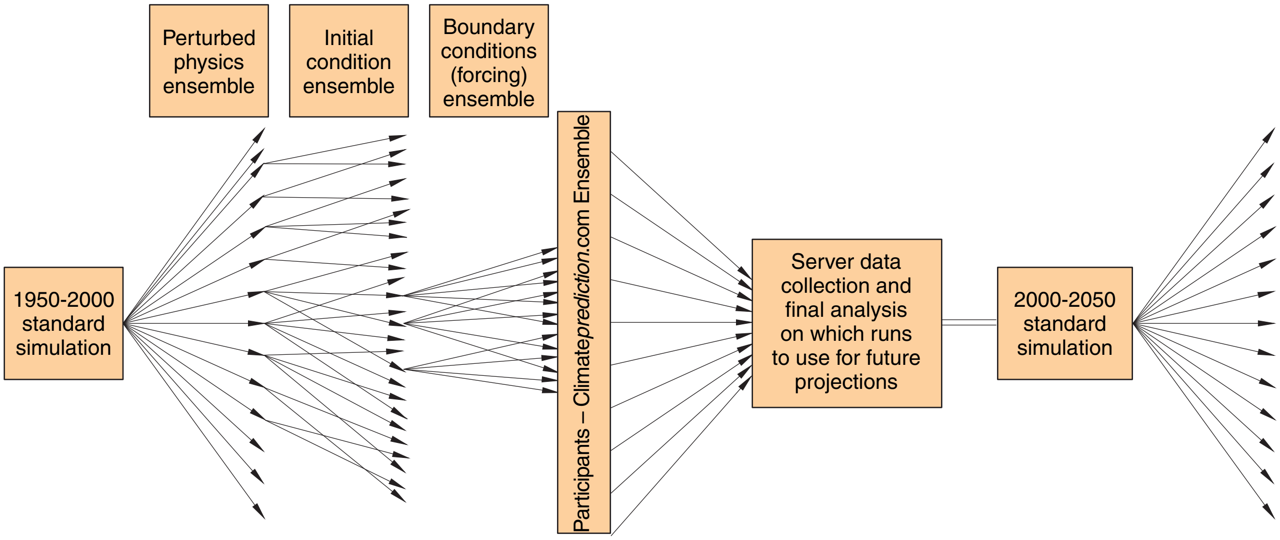

Figure 4. The combination of perturbed physics, initial condition, and boundary condition (greenhouse gas forcing) ensembles leads to a requirement for a large number (O(2M)) of unique simulations.

projects on topics including uncertainty analysis of regional climate change, climate feedbacks, and climate process studies.

Project software design

We initially plan to distribute the bulk of the participant software package from a server over the Internet, but in the medium term, we hope to also make it available through magazine cover CDs. The package will be compressed, of course, and provide a basic installation facility. After installation, the client is run and goes through the following procedures:

1. It gathers the participant's details and contacts a server to register both the individual and the machines he or she is making available.
2. It downloads from the server a small control file (Fortran namelist), which defines the unique aspects of the individual's simulation.
3. It sets the experiment running as a lowest priority process.
4. When the experiment is complete, it uploads a subset of the results back to the server.

During the project's current demonstration phase, we're doing the registration and control file download entirely through Web pages, but we will automate this before public release. In the medium term, the client software will return results while the simulation is still running, but this is not an important issue for the initial release, in which most of the important data is generated toward the end of the simulations.

We hope that the distribution of participants will reflect the global nature of the scientific issue being studied. If so, we will need a series of servers around the world, each one able to distribute the software package, take registrations, distribute unique control files, and take and store returned data. This requires managing a globally distributed database of participants, simulations, and data. It also raises a series of issues regarding the analysis of distributed data sets. There are substantial overlaps here with problems that other Grid and Datagrid projects are addressing.[9]

We designed the software package to include several distinct subpackages, as Figure 5 shows. Specifically, it includes the

- Client, which manages the user interaction
- Participant's simulation, which includes the climate model and experiment control and the resulting data's post-processing and initial analysis
- Uploader, which contacts a server and returns results
- Visualization, which lets participants view their simulation's progress as it develops

We designed the package in a modular fashion to simplify upgrades of separate sections without requiring large downloads. In particular, the visualization is likely to go through a series of different versions as it is expanded to include extra functionality. Because the visualization is not fundamental to running the experiment, it is not part of the basic download but will be made available separately. As a result, the initial download has been kept under 5 Mbytes.

The fundamental difficulty in developing a suitable package has been the breadth and depth of computational expertise required to design and produce simply installable software suitable for running on a range of Windows operating systems with a variety of different setups. The model itself is written in Fortran, the client and uploader in visual C++, the visualization in a specialized visualization language,

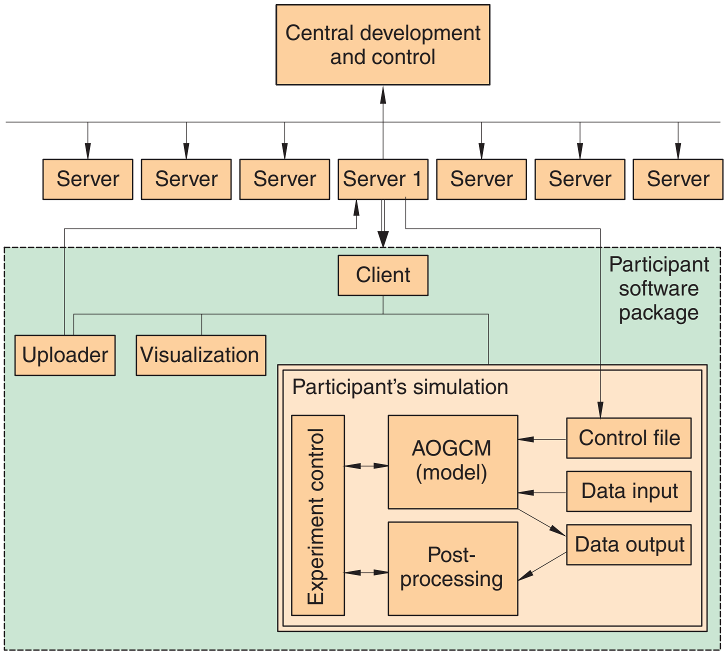

Figure 5. The relationship between the different software in the climate*prediction*.com project.

the uploader uses HTTP to contact the server (which itself requires HTML and Perl to manage the downloads and archiving), and SQL for database control.

Client and model implementation

The Unified Model is designed for implementation on several single-processor and MPP systems running Unix operating systems. It comes with a Tcl/Tk user interface and is extremely flexible in its design, allowing major choices such as whether to include an ocean, an atmosphere, a stratosphere, resolution variations, regional area versions, and so on. The biggest barrier to new implementations is the need to find a suitable compiler that can cope with the code's oddities—much of it was originally developed with Fortran 77, sometimes using nonstandard features specific to a particular compiler. The user interface's flexibility is not needed for climate*prediction*.com because participants won't need to modify their climate model setup. However, we do exploit the flexible design to give us only one executable for the many different model setups.

We have developed a standalone, distributable version of the Linux model to carry out initial tests and ongoing work. The Windows model is based on this Linux version but compiled for Windows using the Compaq Visual Fortran compiler. Under Linux, the control of the experiment—post processing of results, file format conversions, and so on—is done with basic shell scripting; under Windows, we are using Visual C++.

We wrote the client, which carries out most of the user and system interaction, in Visual C++ because it is the language of preference for Windows programmers and simplifies much of the code by including Microsoft Foundation Classes.

A major task was the development of a system to enable the separate visualization executable to access the fields from the model as it is running. We achieved this by having the client set aside an area of "shared" memory that both the model and visualization packages could access.

Servers

One central server will initially manage the ensemble of climate models and the data generated. As the participant base for the project grows and becomes more geographically distributed, we'll need a network of servers. Their primary functions will be

- Management of the database of participants, computers, and climate simulations
- Generation and distribution of the unique climate model control files
- Receiving, checking, and archiving the climate simulation results
- Producing summary analysis and statistics for the Web site
- Making the climate simulation results available for scientific research and academic projects

The server implementation is based on a standard Web server and a mySQL database. All communication between the client on the participant's PC and the server will be via standard HTTP. An important concern is the maintenance of the experiment's integrity, so the server will make careful checks on uploaded files to ensure that each participant returns appropriate data from the run assigned to him or her.

The client will always initiate its communication with the server. This will help maintain security on the participant's computer. Although the client will receive incoming data, it will only be sent when requested and in a specific form. At all other times, the client is deaf and will not accept any incoming data.

Our ongoing work addresses a series of issues including file compression, experiment security, and automatic production of summary analyses and statistics to include on the Web site. Other distributed computing projects have reported substantial interest in the com-

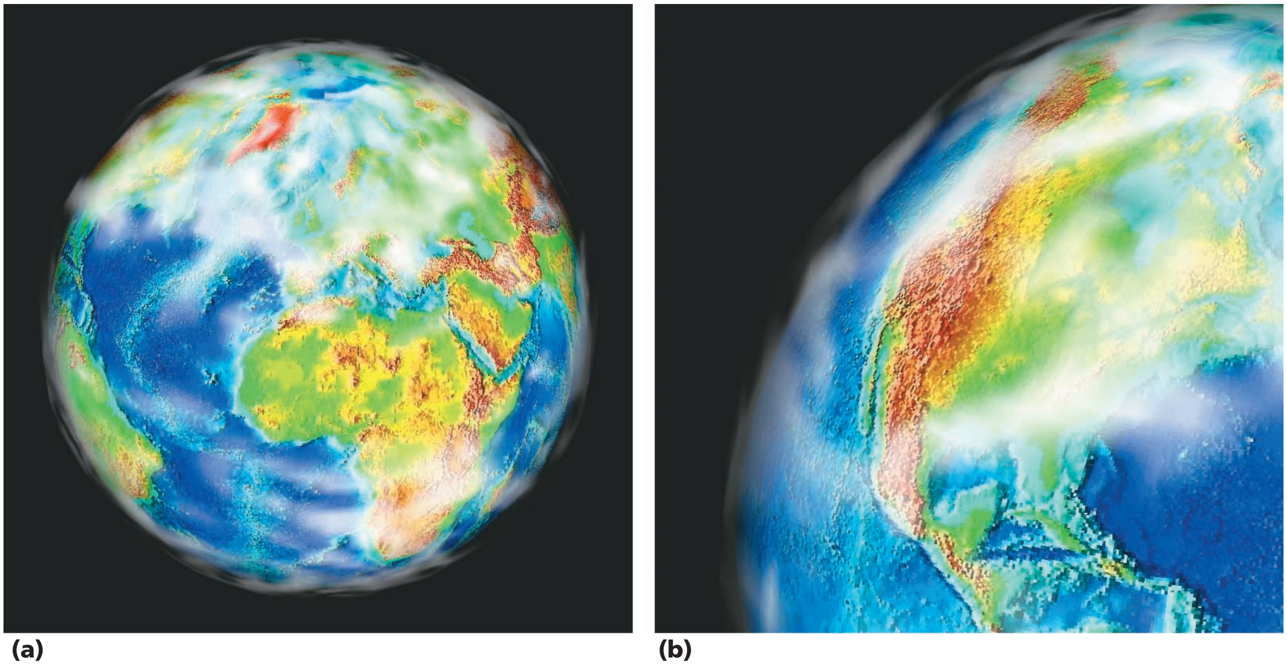

Figure 6. Examples of model visualizations available to participants. (a) A 3D global view of model cloud fields; (b) a "zoomed in" version. Underlying Earth's surface relief image courtesy of NOAA–National Geophysical Data Center.

petitive nature of participation, in terms of how much computing power an individual or team has contributed. In our case, this contribution can be expressed as days or years of simulation.

Data storage and analysis

Data storage and analysis will be a major challenge for the project, which could generate a petabyte of data if the participant level reaches the million or so hoped for. To cope with this volume of data, we need several layers of storage. At the top level, summary diagnostics will be available to everyone on a central server (or replicated over several regional servers). At an intermediate level, a set of climate model fields will be stored on distributed servers. At the bottom layer, for participants with sufficient spare disk space, data will be left on participants' computers.

The data held at the intermediate level on distributed file servers will be made available through thin clients based on those developed at the Program for Climate Model Diagnosis and Intercomparison at the Lawrence Livermore Laboratory as part of its plans for the Earth System Grid II. These clients will provide access to the results for analysis and visualization without having to know the data's details or where it is stored. One challenge is to develop effective metadata to describe and catalog the intermediate-level files, making data mining as efficient as possible.

Accessing data on the lowest layer, participants' PCs, raises many technical problems. The data set will be incomplete because different participants will choose to keep different amounts of data, and its availability will be intermittent because most participants are online only occasionally. The challenge of addressing these issues is large but potentially widely beneficial. Obviously, no essential data will be held at this lowest level, but the data that is held will be useful in further detailed analysis.

Visualization and education

To maintain participant interest, it will be crucial to provide facilities that keep them actively involved in the experiment. To this end, a visualization package will be available with which users can view their simulation's progress on a timestep-by-timestep basis (1 timestep = 30 simulated minutes). Various options will be available including 3D displays (see Figure 6), zooming in on areas of interest, animations of recent fields, graphs of trends over time, and so on. Furthermore, we plan to develop software based on peer-to-peer computing techniques. People will be able to compare their simulations with centrally held simulations and with friends or other participants running a user-specified version of the model.

This, of course, links to the project's large educational potential, and we have received many enquiries from teachers regarding the development of related teaching material. In particular, the visualization and peer-to-peer software will provide substantial opportunities for mini-research projects in schools and undergraduate courses. Indeed, extensions of this software could also simplify research and analysis by climate researchers worldwide.

In the future, we hope to make available additional packages such as simplified climate models. The participants will be able to vary these themselves and use them to study different climate scenarios while comparing the results with those from the full complex model. Furthermore, we plan to provide impact models that link in with the main model to give some indication of the effects on water supply, malarial regions, agricultural production, and so on if the Earth's climate actually were to develop along the lines of the participant's particular model version.

Analysis of the ensemble of simulations as a whole will also raise interesting issues relating to visualization of results in high-dimensional parameter space. Researchers are investigating various ways of addressing this problem.[10]

The climate*prediction*.com project has so far concentrated on producing a package for Windows operating systems, due to their ubiquity. However, we have also produced a limited version for Linux, including only the basic standalone model, which we could easily replicate for OSF 1 and Sun Solaris systems. We have not yet decided whether to release this limited version to the public. The administrative overhead of managing different versions, and the queries they will inevitably stimulate, could prove impossible to cope with unless we find sponsorship. People have also requested versions for other types of machines, in particular Macintoshes, which would be highly desirable. Again, unless support for this activity is forthcoming, it is unfortunately unlikely that the small team currently on this project would be able to pursue this work.

Dave Stainforth is a researcher of climate and atmospheric physics at the University of Oxford. His technical interests include climate research and quantification of uncertainty in climate simulations, e-science and distributed computing, and raising awareness and understanding of climate issues. He has a BA in physics from Oxford University and an MSc in energy and environmental management from Glasgow Caledonian University. Contact him at AOPP, Dept. of Physics, Univ. of Oxford, Clarendon Lab., Parks Rd., Oxford, OX1 3PU, UK; d.stainforth1@physics.ox.ac.uk; www.climateprediction.com

Jamie Kettleborough is a research associate at the British Atmospheric Data Centre at the Rutherford Appleton Laboratory. His technical interests include data management and uncertainty in forecasts of climate change. He has a BSc in chemical physics from the University of Edinburgh and a PhD in atmospheric chemistry from the University of Cambridge. He is a member of the Royal Meteorological Society and the American Geophysical Union. Contact him at SSTD, Rutherford Appleton Lab., Chilton, Didcot, Oxon OX11 0QX, UK; J.A.Kettleborough@rl.ac.uk.

Myles Allen is an NERC advanced research fellow and university lecturer in the Department of Physics, University of Oxford. His technical interests include the dynamics of large-scale climate and climate change. He has a BA in physics and philosophy and a DPhil in physics from the University of Oxford. Contact him at AOPP, Clarendon Lab., Parks Rd., Oxford, OX1 3PU, UK; myles.allen@physics.ox.ac.uk.

Mat Collins is a senior research fellow in the Dept. of Meteorology at the Centre for Global Atmospheric Modeling. His research interests include climate research and distributed computing. He has a BSc in pure and applied mathematics from the University of Swansea and a PhD in meteorology from the University of Reading. He is a fellow of the Royal Meteorological Society. Contact him at the Dept. of Meteorology, Univ. of Reading, Earley Gate, Reading, RG6 6BB, UK; matcollins@met.rdg.ac.uk; www.met.rdg.ac.uk/~mat.

Andy Heaps is a computer system manager in the Department of Meteorology at the Centre for Global Atmospheric Modeling. His technical interests include model porting, performance, and graphics. He has a BSc in physics from York University and an MSc in atmospheric science from the University of East Anglia. Contact him at Dept. of Meteorology, Centre for Global Atmospheric Modeling, Univ. of Reading, Earley Gate, Reading, RG6 6BB, UK; andy@met.rdg.ac.uk.

James Murphy is a research manager in the Climate Change Group at the Hadley Centre for Climate Prediction and Research. His technical interests include ensemble climate prediction on decade to century timescales, quantifying uncertainty in model predictions, and simulations of regional climate. He has a BSc in physics from Manchester University. Contact him at the Hadley Centre for Climate Prediction and Research, Meteorological Office, Bracknell, Berkshire RG12 2SY, UK; james.Murphy@metoffice.com.

References

1. M.R. Allen, "Do-It-Yourself Climate Prediction," *Nature*, vol. 401, no. 6754, Oct. 1999, p. 642.
2. E. Korpela et al., "SETI@home: Massively Distributed Computing for SETI," *Computing in Science & Eng.*, vol. 3, no. 1, Jan./Feb. 2001, pp. 78–83.
3. Working Group I, "Climate Change 2001: The Scientific Basis," *Third Assessment Report of the Intergovernmental Panel on Climate Change*, Cambridge Univ. Press, Cambridge, UK, 2001.
4. C.E. Forest et al., "Quantifying Uncertainties in Climate System Properties Using Recent Climate Observations," *Science*, vol. 295, no. 5552, Jan. 2002, pp. 113–117.
5. M.J.P. Cullen, "The Unified Forecast/Climate Model," *Meteorological Magazine*, vol. 122, no. 1449, Apr. 1993, pp. 81–94.
6. K.D. Williams, C.A. Senior, and J.F.B. Mitchell, "Transient Climate Change in the Hadley Centre Models: The Role of Physical Processes," *J. Climate*, vol. 14, no. 12, Dec. 2001, pp. 2659–2674.
7. C. Gordon et al., "The Simulation of SST, Sea Ice Extents and Ocean Heat Transports in a Version of the Hadley Centre Coupled Model without Flux Adjustments," *Climate Dynamics*, vol. 16, Feb. 2000, pp. 147–168.
8. P.M. Cox et al., "Acceleration of Global Warming Due to Carbon-Cycle Feedbacks in a Coupled Climate Model," *Nature*, vol. 408, no. 6809, Nov. 2000, pp. 184–187.
9. R. Moore et al., "Data-Intensive Computing," *The Grid: Blueprint for a New Computing Infrastructure*, I. Foster and C. Kesselman, eds., Morgan Kaufmann, San Francisco, 1999.
10. J.X. Chen and W. Shuangbao, "Data Visualization: Parallel Coordinates and Dimension Reduction," *Computing in Science & Eng.*, vol. 3, no. 5, Sept./Oct. 2001, pp. 110–C3.