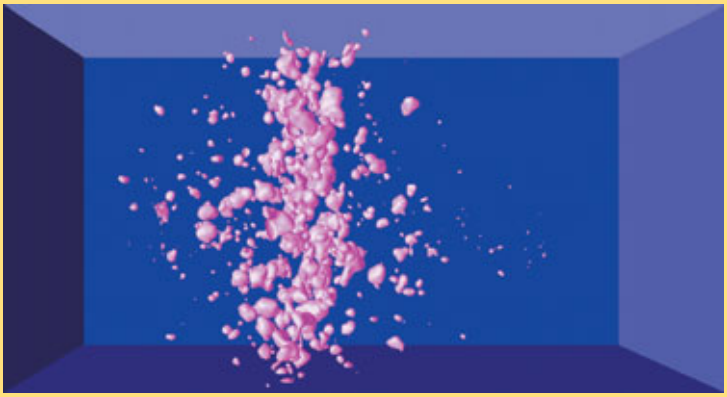
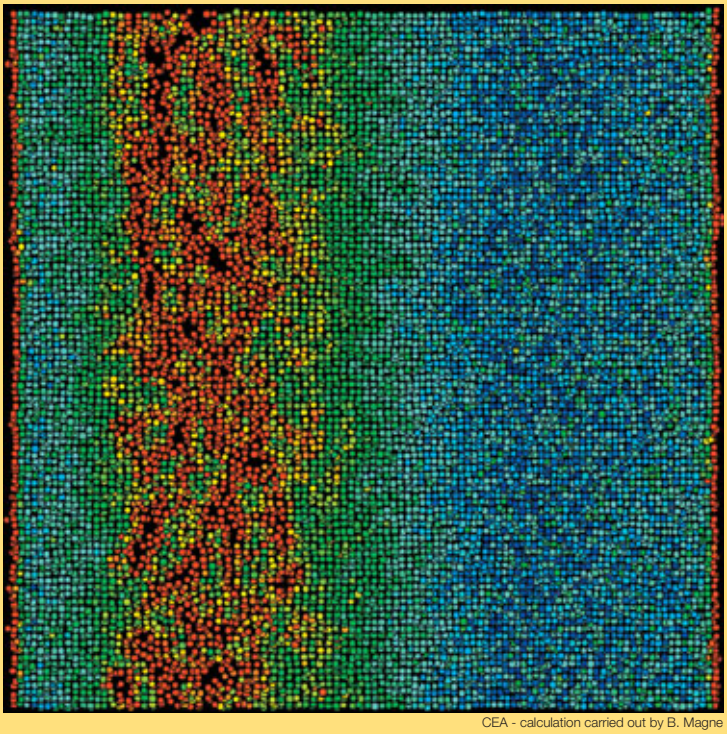
I. SIMULATION FOR UNDERSTANDING

Even more than is the case in technological development, where it has demonstrated its effectiveness, simulation holds a special place in scientific research, since it falls to it to represent the effects of fundamental processes, even though some are insufficiently understood, or even as yet unknown. The various topics addressed in this chapter show how simulation helps in the understanding of the phenomena, but equally how it is no more than one link in the chain of understanding, between theory, modeling and experimentation, the latter remaining the "final arbiter." This is spectacularly the case, for example, in the field of materials, where simulation builds a bridge between the theory of phenomena at atomic scale and observation of macroscopic effects. In climatology, modeling of the ensemble comprising the atmosphere and oceans is essential, in order to predict the Planet's destiny. Nanotechnologies call for modeling of surfaces, interfaces and other nanostructures. Biotechnologies, in the broad sense of the term, can no longer do without molecular modeling. As for nuclear waste, simulation turns out to be indispensable, to predict its very-long-term behavior after disposal in a geological environment and thus to optimize the design of packages, containers and of the site.

Yet another aspect is that "behavioral" modeling makes it possible to ensure that a small number of targeted experiments may be carried out, with no risk of passing over decisive parameters.



CEA - analysis carried out by J. Bontaz-Carion *et al.*



CEA - calculation carried out by B. Magne

Comparison of the microtomography of a copper sample, showing impact damage (top), with the same impact phenomenon simulated by molecular dynamics (bottom). For the analyzed sample, the parallelepiped measures 1,200 µm × 1,200 µm × 2,400 µm. Dimensions for the simulated system are 20 nm × 20 nm × 3.6 nm for 100,000 atoms at normal density. A 20,000-atom section, 0.36 nm thick, is represented here. Duration of the experiment is 7 picoseconds. Pore size distribution for the analyzed sample follows a scale law comparable to that found with molecular dynamics.

MODELING CLIMATE

The Earth's climate is the result of complex interactions between a large number of processes involving the atmosphere, oceans and continental landmasses. How does this system operate? Can its evolution be forecast on the scale of a season, or longer? Are human activities gradually modifying the major climatic balances? What are their current, and future, consequences for humankind? Such questions make research into climate and climate change particularly sensitive, bringing under the spotlight as they do the results of climate models. These numerical models, also referred to as global circulation models (GCMs), make it possible to simulate the evolution over time of the three-dimensional characteristics of the atmosphere and ocean by taking on board their interactions with the continental land masses and ice masses (ice caps and sea ice).

The Eastern region of Southern Africa as observed by the Meris instrument of the Envisat satellite in July 2002. Such an image, on which the ocean, land masses and clouds, large coastal towns, snow on the mountains of Lesotho and the sands of the Kalahari Desert (top left) can all be seen, illustrates the complexity of the climate system.

ESA 2002

The first global circulation **models** date back to the 1960s, gaining in complexity over the years largely as a result of advances in **supercomputers** (see Box 1, and Box B, ***Computational resources for high-performance numerical simulation***). What are these models? How are they used and what are their limitations? The following pages attempt to shed some light on the issue.

Components of a climate model

The investigation of climate requires taking into account the fluid dynamics of oceans and atmosphere, the physical–chemical processes of the chemicals present in them and the complex interactions with the land or marine **biosphere** and cryosphere.(1) Figure 1 shows how each component in the climatic system (atmosphere, ocean, bios-

(1) The cryosphere is the portion of the climate system that includes the ice masses and snow deposits of the whole world (inlandsis, ice shelves, ice caps and glaciers, sea ice, seasonal snow cover, lake and river ice, seasonal land frosts and permafrost).

A recent history

1

Climate simulations, just like weather forecasting, owe their rise to the spectacular growth in computers during the twentieth century. The British author L. F. Richardson, in his treatise *Weather Prediction by Numerical Process*, published in 1922, first showed how trends for atmospheric variables (temperature, winds, etc.) can be calculated at various points in space, by substituting for the differential equations describing atmospheric circulation a set of equations based on algebraic differences. However, it would have taken 64,000 persons to perform those calculations! Not until 1950 did American author J. G. Charney execute the first numerical forecast using a simplified **model** of the atmosphere. In the 1960s, weather forecasting gradually became operational in many countries. Starting in the 1970s with models of the atmosphere, climate simulations have developed constantly, so that over the last decade they have taken in not only the atmosphere, but also its couplings with the other components of the climate system. Major work hinging on ocean–atmosphere coupling was conducted in France starting in the early 1990s, and the European PRISM Program (PRogram for Integrated earth-System Modeling) (2002–2005) has the goal of facilitating interfacing of the various climate-system models available in Europe.

phere and cryosphere) has its own time constants. The atmosphere features the most rapid variations and has little memory. The deep ocean, however, can capture and hold perturbations over hundreds of years. Such different time constants between climate subsystems allow an understanding of the difference between climatology and meteorology. A meteorological forecast consists in determining, for instance, whether it will rain over the next few days. As the ocean varies more slowly than the atmosphere, the forecaster rightly ignores its variations over the forecast period. Investigating climate, however, requires longer-term (10–100 years) simulations if reliable statistics are to be obtained. The purpose now is to reproduce not the system's trajectory, but the statistics for such trajectories. In such a context, the variations in the circulation of the oceans and other, slower-moving water reservoirs can no longer be ignored. The climate system under consideration thus depends on the processes and time scales being investigated. For some years, the coupled ocean–atmosphere system has become central, as these two fluids share in the redistribution of surplus energy, received from the equatorial regions, to the polar regions, through winds and oceanic currents.

A climate model is primarily a physical model, i.e. the physical processes to be represented are expressed as mathematical equations. The main components of the climate system give rise to sub-models, allowing calculation of their individual internal processes and mutual couplings. Thus, the model of the atmosphere, which is basically none other than a weather-forecasting model, calculates wind heat and moisture transport, and

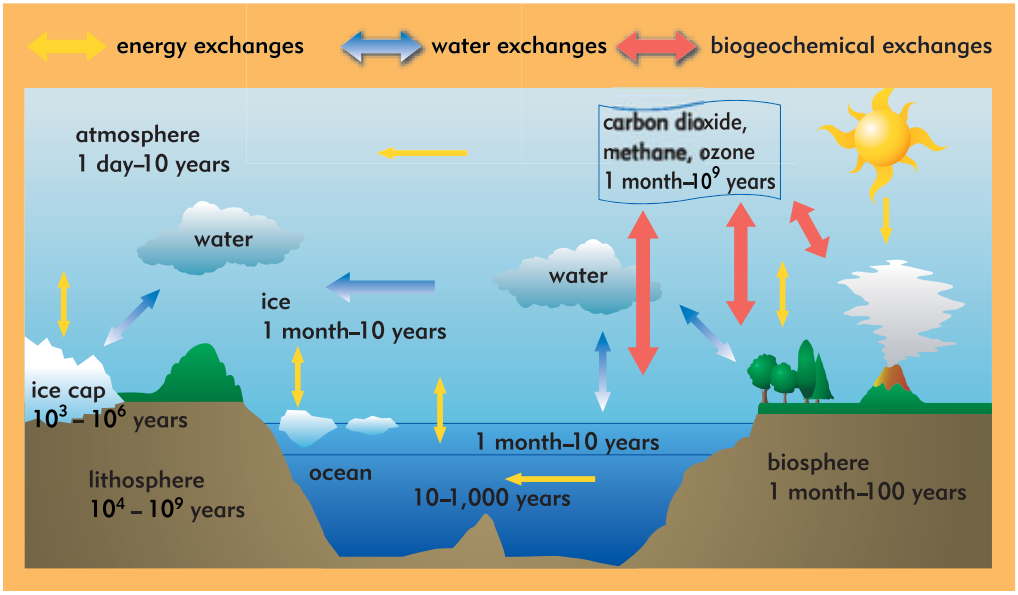


*Figure 1. Components of the climate system according to Sylvie Joussaume (*Climat d'hier à demain*, CNRS Éditions, Paris, 1993).*

Computational resources for high-performance numerical simulation

B

Carrying out more accurate **numerical simulations** requires the use of more complex physical and numerical **models** applied to more detailed descriptions of the simulated objects (see Box A, ***What is a numerical simulation?***). All this requires advances in the area of simulation software but also a considerable increase in the capacity of the computer systems on which the software runs.

Scalar and vector processors

The key element of the computer is the processor, which is the basic unit that executes a program to carry out a computation. There are two main types of processors, **scalar processors** and **vector processors**. The former type carries out operations on elementary (scalar) numbers, for instance the addition of two numbers. The second type carries out operations on arrays of numbers (vectors), for example adding elementwise the numbers belonging to two sets of 500 elements. For this reason, they are particularly well suited to numerical simulation: when executing an operation of this type, a vector processor can operate at a rate close to its maximum (peak) performance. The same operation with a scalar processor requires many independent operations (operating one vector element at a time) executed at a rate well below its peak rate. The main advantage of scalar processors is their price: these are general-purpose microprocessors whose design and production costs can be written-down across broad markets.

Strengths and constraints of parallelism

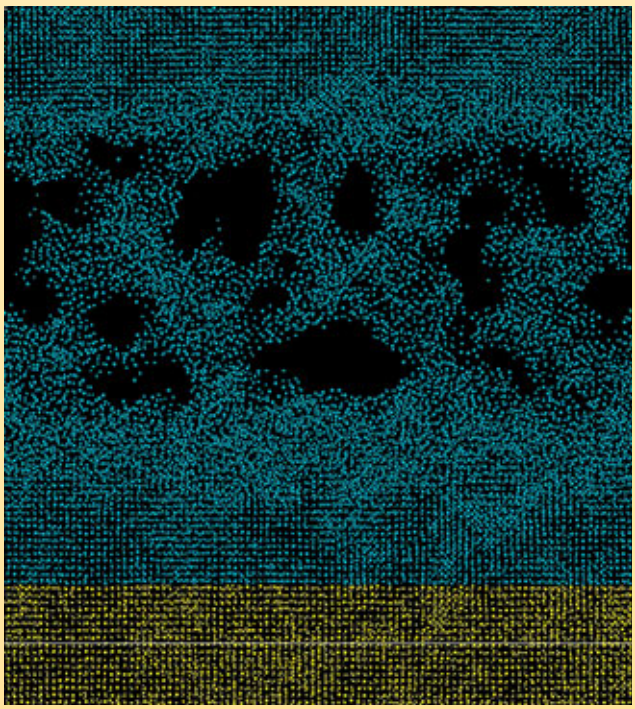
Recent computers allow high performances partly by using a higher operating frequency, partly by trying to carry out several operations simultaneously: this is a first level of **parallelism**. The speeding up in frequency is bounded by developments in microelectronics technology, whereas interdependency between the instructions to be carried out by the processor limits the amount of parallelism that is possible. Simultaneous use of several processors is a second level of parallelism allowing better performance, provided programs able to take advantage of this are available. Whereas parallelism at processor level is automatic, parallelism *between processors* in a parallel computer must be taken into account by the programmer, who has to split his program into independent parts and make provisions for the necessary communication between them. Often, this is done by partitioning the domain on which the computation is done. Each processor simulates the behavior of one domain and regular communications between processors ensure consistency for the overall computation. To achieve an efficient parallel program, a balanced share of the workload must be ensured among the individual processors and efforts must be made to limit communications costs.

The various architectures

A variety of equipment types are used for numerical simulation. From their desktop computer where they prepare computations and analyze the results, users access shared computation, storage and visualization resources far more powerful than their own. All of these machines are connected by networks, enabling information to circulate between them at rates compatible with the volume of data produced, which can be as much as 1 **terabyte** (1 TB = 10^{12} bytes) of data for one single simulation.
The most powerful computers are generally referred to as **supercomputers**. They currently attain capabilities counted in **teraflops** (1 Tflops = 10^{12} floating-point operations per second).
Currently, there are three main types of supercomputers: vector supercomputers, clusters of mini-computers with shared memory, and clusters of PCs (standard home computers). The choice between these architectures largely depends on the intended applications and uses. Vector supercomputers have very-high-performance processors but it is difficult to increase their computing performance by adding processors. PC clusters are inexpensive but poorly suited to environments where many users perform numerous large-scale computations (in terms of memory and input/output).
It is mainly for these reasons that CEA's Military Applications Division (DAM) has choosen for its Simulation Program (see *The Simulation Program: weapons assurance without nuclear testing*) architectures of the shared-memory mini-computer cluster type, also known as **clusters of SMPs** (symmetric multiprocessing). Such a system uses as a basic building block a mini-computer featuring several microprocessors sharing a common memory (see Figure). As these mini-computers are in widespread use in a variety of fields, ranging from banks to web servers through design offices, they offer an excellent performance/price ratio. These basic "blocks" (also known as *nodes*) are connected by a high-per-

Installed at CEA (DAM-Ile de France Center) in December 2001, the TERA machine designed by Compaq (now HP) has for its basic element a mini-computer with 4 x 1-GHz processors sharing 4 GB of memory and giving a total performance of 8 Gflops. These basic elements are interconnected through a fast network designed by Quadrics Ltd. A synchronization operation across all 2,560 processors is completed in under 25 microseconds. The overall file system offers 50 terabytes of storage space for input/output with an aggregate bandwidth of 7.5 GB/s.

Computational resources for high-performance numerical simulation (cont'd)



CEA

Parallel computers are well suited to numerical methods based on meshing (see Box A, What is a numerical simulation?) but equally to processing ab-initio calculations such as this molecular-dynamics simulation of impact damage to two copper plates moving at 1 km/s (see Simulation of materials). The system under consideration includes 100,000 atoms of copper representing a square-section (0.02 µm square) parallelogram of normal density. The atoms interact in accordance with an embedded atom potential over approximately 4–6 picoseconds. The calculation, performed on 18 processors of the Tera supercomputer at Bruyères-le-Châtel using the CEA-developed Stamp software, accounted for some ten minutes of "user" time (calculation carried out by B. Magne). Tests involving up to 64 million atoms have been carried out, requiring 256 processors over some one hundred hours.

formance network: the cumulated power of several hundreds of these "blocks" can reach several Tflops. One then speaks of a **massively parallel computer**.

Such power can be made available for one single parallel application using all the supercomputer's resources, but also for many independent applications, whether parallel or not, each using part of the resources.

While the characteristic emphasized to describe a supercomputer is usually its computational power, the input/output aspect should not be ignored. These machines, capable of running large-scale simulations, must have storage systems with suitable capacities and performance. In clusters of SMPs, each mini-computer has a local disk space. However, it is not advisable to use this space for the user files because it would require the user to move explicitly his data between each distinct stage of his calculation. For this reason, it is important to have disk space accessible by all of the mini-computers making up the supercomputer. This space generally consists in sets of disk drives connected to nodes whose main function is to manage them. Just as for computation, parallelism of input/output allows high performance to be obtained. For such purposes, parallel overall file systems must be implemented, enabling rapid and unrestricted access to the shared disk space.

While they offer considerable computational power, clusters of SMPs nevertheless pose a number of challenges. Among the most important, in addition to programming simulation software capable of using efficiently a large number of processors, is the development of operating systems and associated software tools compatible with such configurations, and fault-tolerant.

François Robin
Military Applications Division
CEA, DAM–Ile de France Center

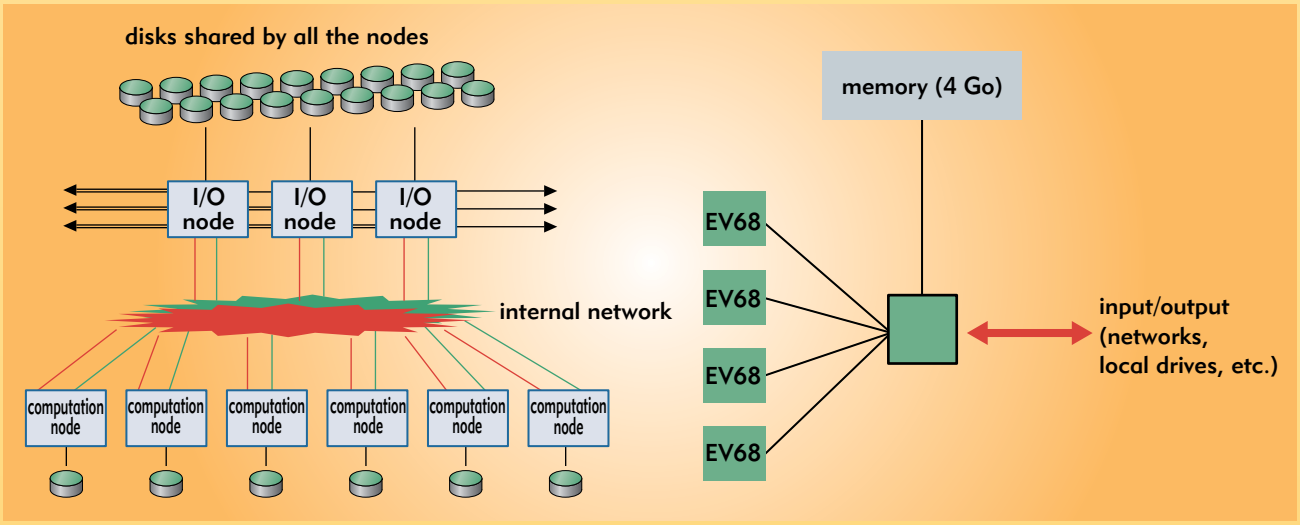
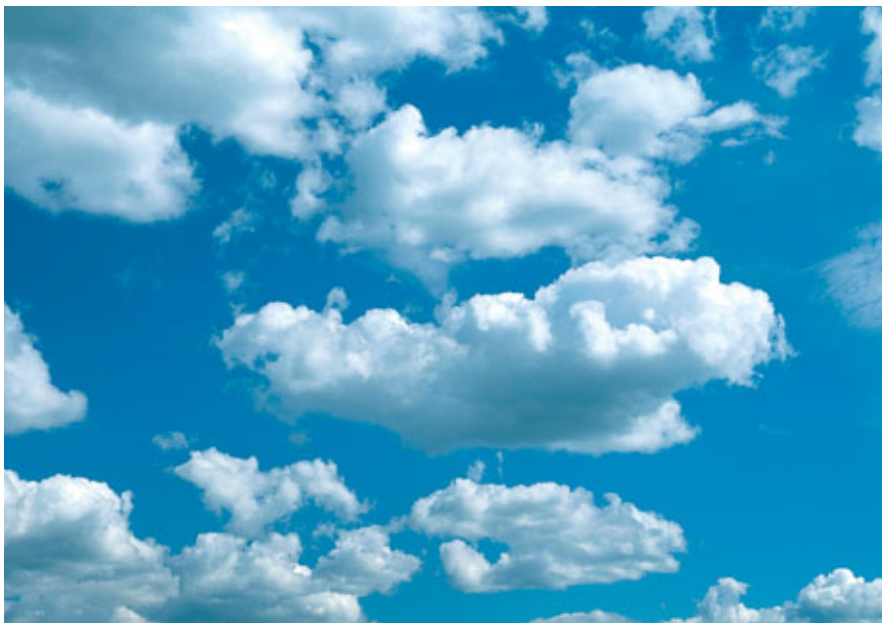


Figure. Architecture of an "SMP-cluster" type machine. At left, the general architecture (I/O = input/output), on the right, that of a node with four Alpha EV68 processors, clocked at 1 GHz.



Clouds are one of the more difficult factors to express in equations for climate modeling.

PhotoDisc

exchanges of momentum, heat and water between the atmosphere and ocean and land surfaces. To this must be added the condensation of moisture in clouds and rainfall, absorption and diffusion of incoming solar radiation, as well as the emission and absorption of infrared radiation by clouds, the atmosphere's gases and ocean and land surfaces. The various factors liable to modify the basic processes are also introduced. Sea ice, snow and plant cover, for instance, all affect the amount of solar radiation absorbed by surfaces, through their ability to reflect incoming solar radiation into space. The model of the ocean takes into account interactions with the atmosphere through winds, heat fluxes and solar and infrared radiation. **Convective** displacements in the water column are determined by the variations in temperature and salinity.

The art of putting the world inside a computer

The term climate "model" can be misleading, for after the physical model comes the numerical model. Putting the Earth inside a computer requires solving equations over a three-dimensional grid on a global scale (see Figure 2). The **numerical methods** used for climate must conserve integral quantities such as energy or total water mass. At the **resolutions** generally used for climate, the equations are solved for every half-hour point over one integration of the model. Many processes, such as those involving clouds, are active at a scale much smaller than that of the **mesh** or time step of the model, and cannot therefore be modeled explicitly. Their effects at the scales represented are taken on

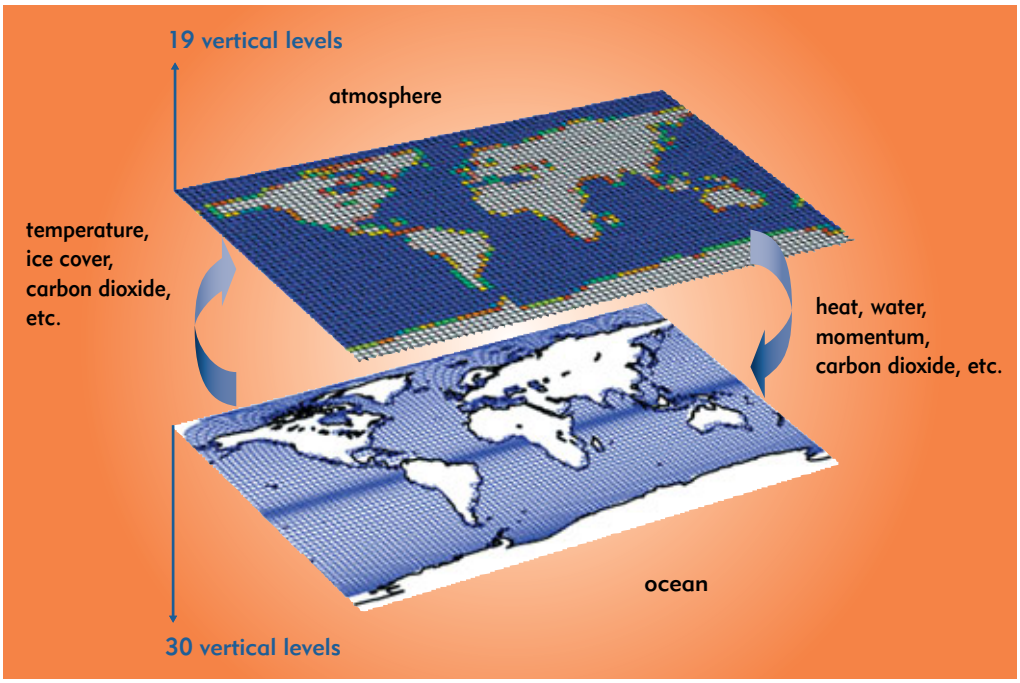


Figure 2. Meshes of the oceanic and atmospheric components of the IPSL (Pierre-Simon-Laplace Institute) coupled model in a so-called "low-resolution" version. Over one integration, coupling of the two models takes place once per simulated day, by means of a coupler managing the communication between models and interpolations between meshes. To ensure energy conservation from one mesh to the other, atmosphere mesh cells may feature various surface types (land, ocean, sea ice and continental ice) and the coastline is obtained by projecting the land/sea mask of the ocean model onto the atmosphere grid. Colors on the atmosphere mesh show the percentage of land in each cell. The ocean mesh has a finer resolution in latitude towards the equator to yield better representation of tropical phenomena.

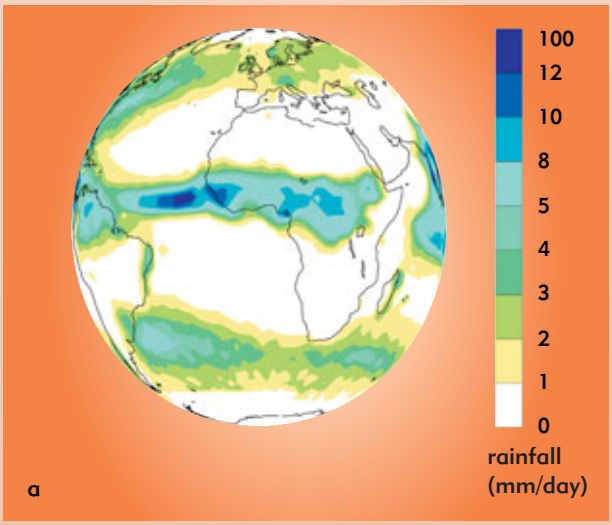
How to validate models and assess their accuracy?

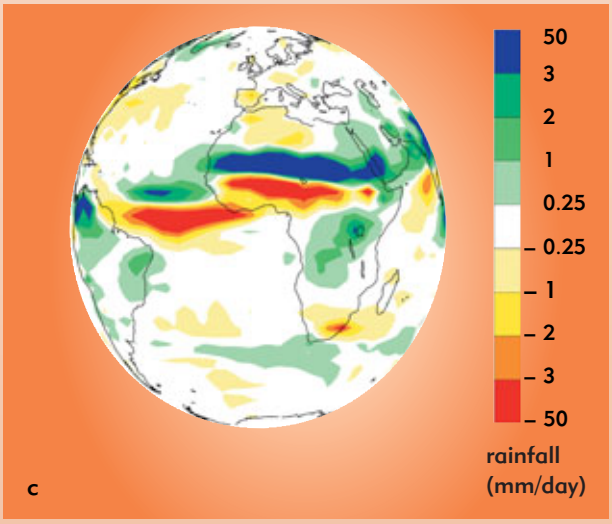
The validity of the results depends on a number of factors. The first bears on the physical processes reproduced in the **model**. This factor alone is not sufficient, and a whole battery of tests have to be carried out to verify the simulated climate is in agreement with the observed phenomena. The many chaotic aspects of the climate system (their evolution is highly sensitive to even small perturbations in initial conditions and boundary conditions) make a recourse to ensembles of simulations imperative, or require that findings be based on the results of a number of models. Thus, many projects comparing the results from models of current climate or past climates have shown that the models satisfactorily represented the main features of climate and its variations, but that no one model was presently capable of reproducing correctly all the facets of climate. One of the main sources of uncertainty relates to the representation of clouds, which largely conditions the energy response of the models to a perturbation.

Simulations of past climates also provide an opportunity to assess whether the models are capable of reproducing climates different from the current climate. This is one of the goals of the international PMIP Program (Paleoclimate Modeling Intercomparison Project) coordinated by Sylvie Joussaume at LSCE. The results obtained for the middle Holocene (6,000 years ago) have shown, for instance, that the coupled ocean–atmosphere–vegetation system should be taken as a whole to reproduce the wetter conditions revealed by lake sediments and pollen deposits in regions that are dry nowadays (see Figure).

CNRS Photo Library – Nicole Petit-Maire

These lake sediments found in the Sahara show that the climate of this region was wetter in the past than it is today.





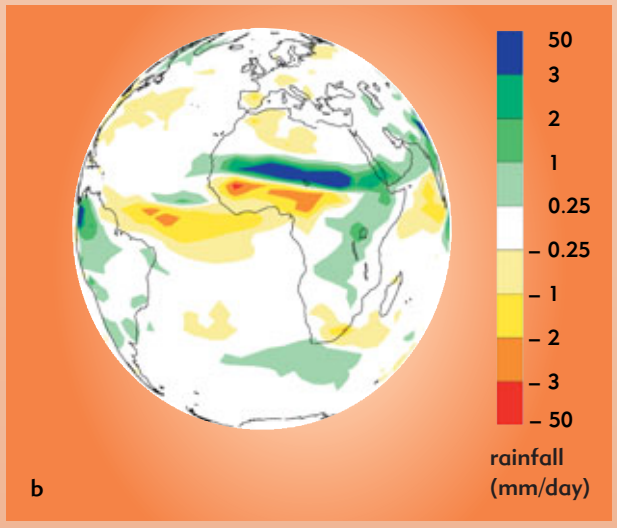
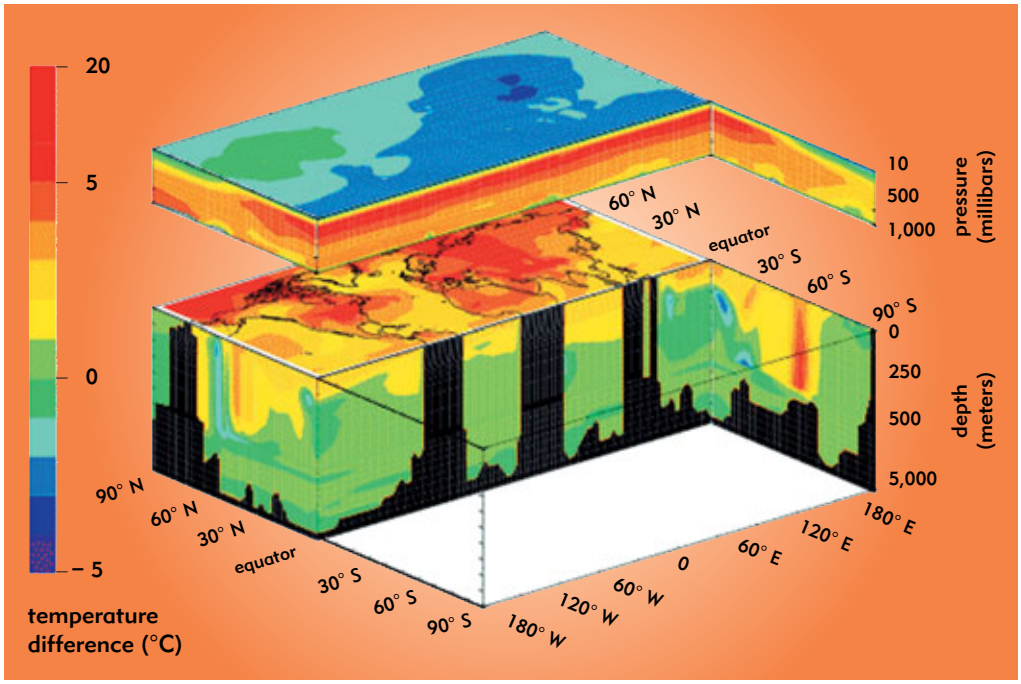


Figure. In summer, the continents of the Northern hemisphere warm up more than the ocean, which has a higher heat capacity. Warm air has a lower density, which is characteristically evidenced by development of low pressure over the continent, the minimum being situated at the foot of the Himalayan range and the Tibetan Plateau and extending as far as the Sahara. The winds are thus drawn from the ocean towards the continent. They increase their water content during their transit across the warm tropical oceans, this being released in the form of torrential rain on the continent. These are the monsoon rains, as shown here in Africa in July–August (a). 6,000 years ago the Earth received more solar radiation in summer in the Northern hemisphere, which reinforced the land–ocean gradient and energized monsoon circulation. Monsoon winds converged further north in Africa, causing precipitation in regions that are dry today. Models of the atmosphere subjected to the solar radiation of 6,000 years ago all show this mechanism (b). These simulations ignore the changes in ocean circulation and vegetation. Simulations carried out with the coupled ocean–atmosphere–vegetation model of the IPSL (Pierre-Simon-Laplace Institute) show that the feedbacks related to ocean and vegetation amplify the response of monsoon circulation (c). The results of this simulation are in better agreement with the available data.



L. Fairhead (CNRS/LMD)

Figure 3. Three-dimensional representation of temperature change simulated with the IPSL (Pierre-Simon-Laplace Institute) climate model for a climate disturbed by man-made carbon dioxide (CO_2) after 240 years. The reference corresponds to a pre-industrial climate, CO_2 content being fixed at its level in the 1860s (180 ppm). The top portion of the figure refers to the atmosphere and the lower portion to the ocean. Isolines are entered at 1 °C intervals (line 0 is omitted) up to 5 °C, at 5 °C intervals thereafter.

board by modeling them on the basis of physical relations with large-scale variables. This technique is known as **parameterization**. Next comes the computer model, or more prosaically the "**code**". FORTRAN is currently widely used for legacy reasons and because it generally performs better than other languages on the **vector machines** on which models of such types are run. The choice of resolution and complexity for the model is also a compromise between the "sought-for realism" and IT costs. Computation must be faster, much faster, than evolved time. For the investigations of past climates carried out at the joint CEA–CNRS Laboratory of Climate and Environment Sciences (CEA–CNRS/LSCE), a coupled ocean–atmosphere model is used with an atmospheric component set at a resolution of 72 points in longitude, 45 points in latitude and 19 vertical levels. At this resolution, its takes 200 hours of processing on the most powerful vector computers to simulate 100 years' climate. This calculation time is multiplied by 10 if a gain in spatial resolution by a factor of 2 in all three directions is to be achieved. Such simulations can be run on a workstation, but a tenfold increase in rendering time is to be expected.

Complementing the simulation code comes an array of tools dedicated to management of simulations over several hundreds or thousands of computer hours. Powerful storage resources and analysis tools allow data from these number-crunching machines to be processed. Graphic analysis of the results and statistical analyses, ranging from simple averages to methods that enable the characterization of climate situations and their occurrence over time, are routinely used in climatology (see Box 2).

Images of the climate of tomorrow

Climate models have many applications, such as the investigation of climatic processes, climate fluctuations from one year to the next, or over decades, or of past or future climate changes. To study climate changes due to human activities, the procedure consists first in setting up a simulation to represent a climate undisturbed by man. A simulation of the disturbed climate is then carried out by modifying, say, atmospheric concentration of carbon dioxide (CO_2), in accordance with a scenario suggested by economists.

Figure 3 shows the global warming found for a scenario of this type for 2100 in a simulation effected with the climate model developed by the Pierre-Simon-Laplace Institute (IPSL, Paris). Compared to the simulation of pre-industrial climate, the continents in the Northern hemisphere show marked warming. At higher latitudes, snow cover has shrunk. Bare ground reflects incoming solar radiation less than snow does, thus enhancing warming of continental surfaces. The ocean, owing to its greater heat capacity than that of land, shows a lesser extent of warming. Warming did not reach the ocean deeps. The vertical stability of the ocean increases, leading to reduced vertical exchanges and modifying the ocean's climatic role. The increased greenhouse effect in the low atmosphere insulates the very high atmosphere, which becomes cooler. The changes in the water cycle associated to such a warming of the climate are still poorly understood. The hydrological cycle involves small-scale processes, which are difficult to represent in the models, and the range of results from various models is much broader than that obtained for temperature change.

On the basis of this type of simulation, by combining a variety of approaches involving simulations of current climate, future climate and past climates, the details of the many mechanisms at work in the climate machine are gradually becoming clear. Climate modeling is currently undergoing tremendous and rapid expansion. It is now possible to couple climatic mechanisms with biogeochemical processes. Climatologists no longer seek merely to investigate the impact of changes in the biogeochemical cycles on climate, but equally to examine the impact of climate on those cycles, which will very certainly, in years to come, alter the way they look at the world, and raise many questions yet as to the most appropriate methods to put such physics inside computers.

Pascale Braconnot and **Olivier Marti**
Laboratory of Climate and
Environment Sciences
CEA–CNRS
Physical Sciences Division
Saclay

What is a numerical simulation? A

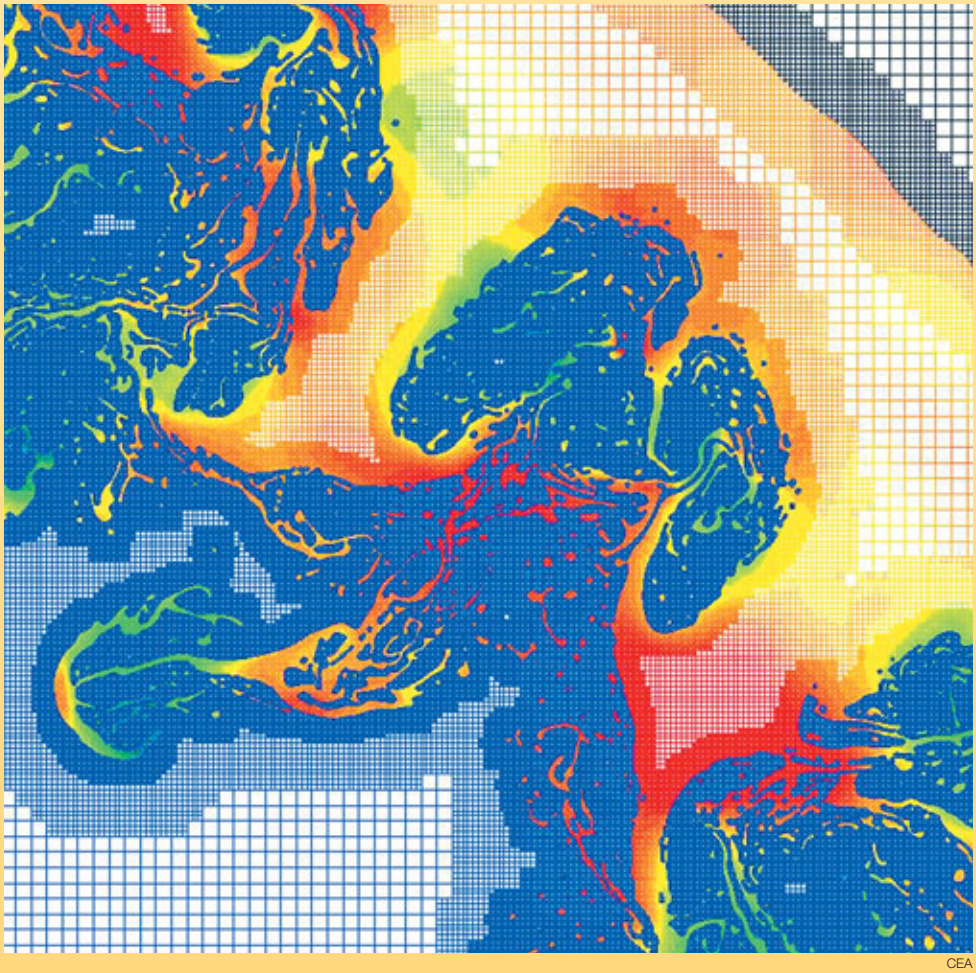
Numerical simulation consists in reproducing, through computation, a system's operation, described at a prior stage by an ensemble of **models**. It relies on specific mathematical and computational methods. The main stages involved in carrying out an investigation by means of numerical simulation are practices common to many sectors of research and industry, in particular nuclear engineering, aerospace or automotive.

At every point of the "object" considered, a number of physical quantities (velocity, temperature…) describe the state and evolution of the system being investigated. These are not independent, being linked and governed by **equations**, generally **partial differential** equations. These equations are the expression in mathematical terms of the physical laws modeling the object's behavior. Simulating the latter's state is to determine – at every point, ideally – the numerical values for its parameters. As there is an infinite number of points, and thus an infinite number of values to be calculated, this goal is unattainable (except in some very special cases, where the initial equations may be solved by analytical formulae). A natural approximation hence consists in considering only a finite number of points. The parameter values to be computed are thus finite in number, and the operations required become manageable, thanks to the computer. The actual number of points processed will depend, of course, on computational power: the greater the number, the better the object's description will ultimately be. The basis of parameter computation, as of numerical simulation, is thus the reduction of the infinite to the finite: **discretization**.

How exactly does one operate, starting from the model's mathematical equations? Two methods are very commonly used, being representative, respectively, of **deterministic computation** methods, resolving the equations governing the processes investigated after discretization of the variables, and methods of **statistical** or **probabilistic calculus**.

The principle of the former, known as the **finite-volume method**, dates from before the time of computer utilization. Each of the object's points is simply assimilated to a small elementary volume (a cube, for instance), hence the *finite-volume* tag. Plasma is thus considered as a set or lattice of contiguous volumes, which, by analogy to the makeup of netting, will be referred to as a **mesh**. The parameters for the object's state are now defined in each mesh cell. For each one of these, by reformulating the model's mathematical equations in terms of volume averages, it will then be possible to build up *algebraic relations* between the parameters for one cell and those of its neighbors. In total, there will be as many relations as there are unknown parameters, and it will be up to the computer to resolve the *system* of relations obtained. For that purpose, it will be necessary to turn to the techniques of **numerical analysis**, and to program specific **algorithms**.

The rising power of computers has allowed an increasing fineness of discretization, making it possible to go from a few tens of cells in the 1960s to several tens of thousands in the 1980s, through to millions in the 1990s, and up to some ten billion cells nowadays (Tera machine at CEA's Military Applications Division), a figure that should increase tenfold by the end of the decade.



CEA

Example of an image from a 2D simulation of instabilities, carried out with CEA's Tera supercomputer. Computation involved adaptive meshing, featuring finer resolution in the areas where processes are at their most complex.

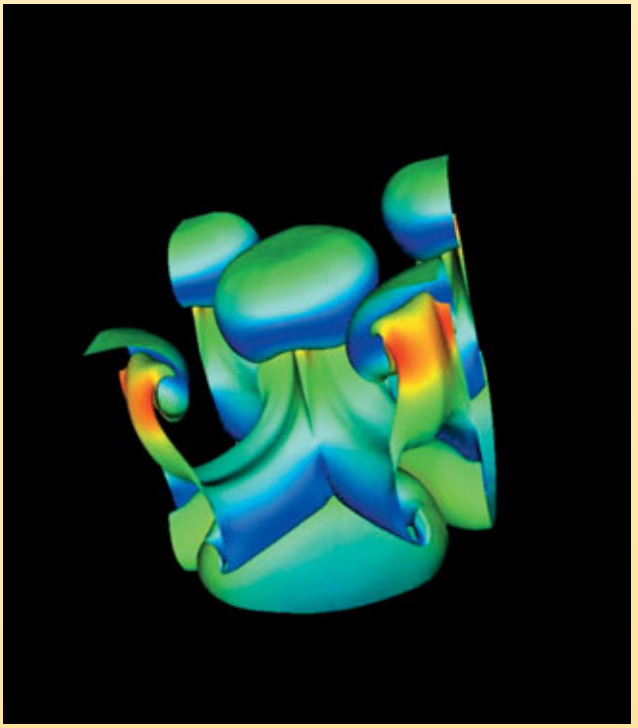
A refinement of meshing, **adaptive remeshing**, consists in adjusting cell size according to conditions, for example by making them smaller and more densely packed at the interfaces between two environments, where physical processes are most complex, or where variations are greatest.

The finite-volume method can be applied to highly diverse physical and mathematical situations. It allows any shape of mesh cell (cube, hexahedron, tetrahedron…), and the mesh may be altered in the course of computation, according to geometric or physical criteria. Finally, it is easy to implement in the context of **parallel computers** (see Box B, ***Computational resources for high-performance numerical computation***), as the mesh may be subjected to partitioning for the purposes of computation on this type of machine (example: Figure B).

Also included in this same group are the **finite-difference method**, a special case of the finite-volume method where cell walls are orthogonal, and the **finite-element method**, where a variety of cell types may be juxtaposed.

The second major method, the so-called **Monte Carlo** method, is particularly suited to the simulation of *particle transport*, for example of neutrons or photons in a **plasma** (see *Simulations in particle physics*). This kind of transport is in fact characterized by a succession of stages, where each particle may be subject to a variety of events (diffusion, absorption, emission…) that are possible *a priori*. Elementary probabilities for each of these events are known individually, for each particle.

It is then a natural move to assimilate a point in the plasma to a particle. A set of particles, finite in number, will form a representative sample of the infinity of particles in the plasma, as for a statistical survey. From one stage to the next, the sample's evolution will be determined by random draws (hence the method's name). The effectiveness of the method, implemented in Los Alamos as early as the 1940s, is of course dependent on the statistical quality of the random draws. There are, for just this purpose, *random-number* methods available, well suited to computer processing.



CEA

3D simulation carried out with the Tera supercomputer, set up at the end of 2001 at CEA's DAM-Île de France Center, at Bruyères-le-Châtel (Essonne* département*).

Finite-volume and Monte Carlo methods have been, and still are, the occasion for many mathematical investigations. These studies are devoted, in particular, to narrowing down these methods' convergence, i.e. the manner in which approximation precision varies with cell or particle number. This issue arises naturally, when confronting results from numerical simulation to experimental findings.

How does a numerical simulation proceed?

Reference is often made to *numerical experiments*, to emphasize the analogy between performing a numerical simulation and carrying out a physical experiment.

In short, the latter makes use of an experimental setup, configured in accordance with initial conditions (for temperature, pressure…) and control parameters (duration of the experiment, of measurements…). In the course of the experiment, the setup yields measurement points, which are recorded. These records are then analyzed and interpreted.

In a numerical simulation, the experimental setup consists in an ensemble of computational programs, run on computers. The **computation codes**, or **software** programs, are the expression, via numerical algorithms, of the mathematical formulations of the physical models being investigated. Prior to computation, and subsequent to it, *environment software* programs manage a number of complex operations for the preparation of computations and analysis of the results.

The initial data for the simulation will comprise, first of all, the delineation of the computation domain – on the basis of an approximate representation of the geometric shapes (produced by means of drafting and CAD [computer-assisted design] software) –, followed by discretization of this computation domain over a mesh, as well as the values for the physical parameters over that mesh, and the control parameters to ensure proper running of the programs… All these data (produced and managed by the environment software programs) will be taken up and verified by the codes. The actual results from the computations, i.e. the numerical values for the physical parameters, will be saved on the fly. In fact, a specific protocol will structure the computer-generated information, to form it into a numerical database.

A complete protocol organizes the electronic exchange of required information (dimensions, in particular) in accordance with predefined formats: modeler,(1) mesher,(2) mesh partitioner, com-

(1) The modeler is a tool enabling the generation and manipulation of points, curves and surfaces, for the purposes, for example, of mesh generation.
(2) The geometric shapes of a mesh are described by sets of points connected by curves and surfaces (Bézier curves and surfaces, for instance), representing its boundaries.

putation codes, visualization and analysis software programs. *Sensitivity* studies regarding the results (sensitivity to meshes and models) form part of the numerical "experiments."
On completion of computation (numerical resolution of the equations describing the physical processes occurring in each cell), analysis of the results by specialists will rely on use of the numerical database. This will involve a number of stages: selective extraction of data (according to the physical parameter of interest) and visualization, and data extraction and transfer for the purposes of computing and visualizing diagnostics.
This parallel between performing a computation case for a numerical experiment and carrying out a physical experiment does not end there: the numerical results will be compared to the experimental findings. This comparative analysis, carried out on the basis of standardized quantitative criteria, will make demands on both the experience and skill of engineers, physicists, and mathematicians. Its will result in further improvements to physical models and simulation software programs.

Bruno Scheurer
Military Applications Division
CEA DAM-Ile de France Center

Frederic Ducros and **Ulrich Bieder**
Nuclear Energy Division
CEA Grenoble Center

The example of a thermalhydraulics computation

Implementation of a numerical simulation protocol may be illustrated by the work carried out by the team developing the **thermalhydraulics** computation software Trio U. This work was carried out in the context of a study conducted in collaboration with the French Radiological Protection and Nuclear Safety Institute (IRSN: Institut de radioprotection et de sûreté nucléaire). The aim was to obtain very accurate data to provide engineers with wall heat-stress values for the components of a pressurized-water reactor in case of a major accident involving turbulent natural circulation of hot gases. This investigation requires simultaneous modeling of large-scale "system" effects and of small-scale **turbulent** processes (see Box F, ***Modeling and simulation of turbulent flows***).
This begins with specification of the overall computation model (Figure A), followed by production of the CAD model and corresponding mesh with commercial software programs (Figure B). Meshes of over five million cells require use of powerful graphics stations. In this example, the mesh for a steam generator (Figures C and D) has been partitioned to parcel out computation over eight processors on one of CEA's parallel computers: each color stands for a zone assigned to a specific processor. The computations, whose boundary conditions are provided by way of a "system" computation (Icare–Cathare), yield results which it is up to the specialists to interpret. In this case, visualization on graphics stations of the instantaneous values of the velocity field show the impact of a hot plume on the steam generator's tube-plate (section of the velocity field, at left on Figure E), and instantaneous temperature in the water box (at right).

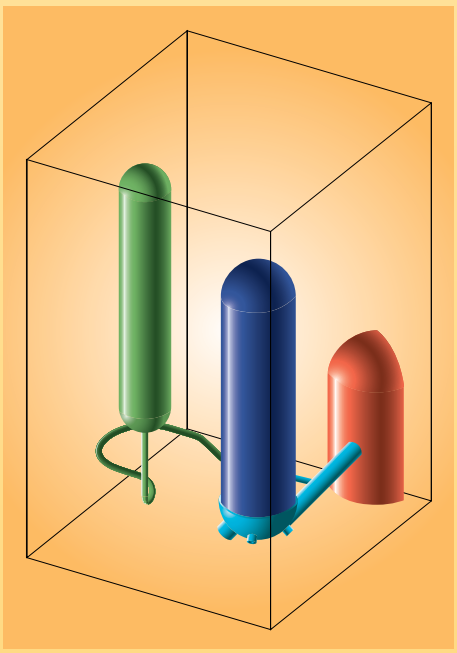


Figure A. Overall computation domain, including part of the reactor vessel (shown in red), the outlet pipe (hot leg, in light blue), steam generator (dark blue), and pressurizer (green).

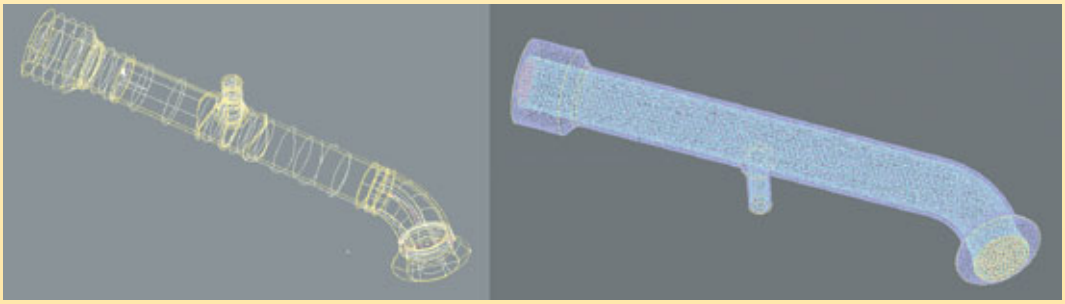
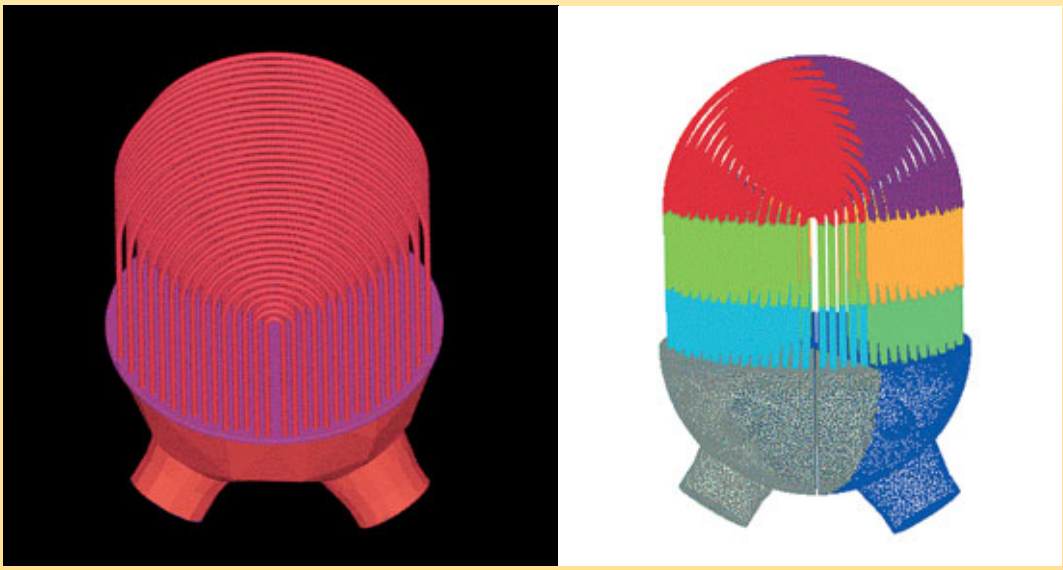


Figure B. CAD model of the hot leg of the reactor vessel outlet (left) and unstructured mesh for it (right).



Figures C and D.

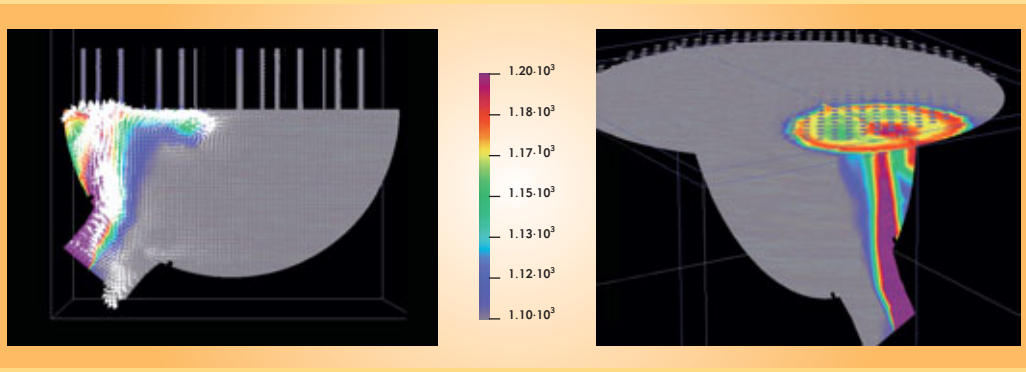


Figure E.

Modeling and simulation of turbulent flows F

Turbulence, or disturbance in so-called turbulent flow, develops in most of the flows that condition our immediate environment (rivers, ocean, atmosphere). It also turns out to be one, if not the, dimensioning parameter in a large number of industrial flows (related to energy generation or conversion, aerodynamics, etc.). Thus, it is not surprising that a drive is being launched to achieve prediction for the process – albeit in approximate fashion as yet – especially when it combines with complicating processes (stratification, combustion, presence of several phases, etc.). This is because, paraxodically, even though it is possible to predict the turbulent nature of a flow and even, from a theoretical standpoint, to highlight certain common – and apparently universal – characteristics of turbulent flows,(1) their prediction, in specific cases, remains tricky. Indeed, it must take into account the considerable range of space and time scales(2) involved in any flow of this type.

Researchers, however, are not without resources, nowadays, when approaching this problem. First, the equations governing the evolution of turbulent flows over space and time (Navier–Stokes equations(3)) are known. Their complete solution, in highly favorable cases, has led to predictive descriptions. However, systematic use of this method of resolution comes up against two major difficulties: on the one hand, it would require complete, simultaneous knowledge of all variables attached to the flow, and of the forced-flow conditions imposed on it,(4) and, on the other hand, it would mobilize computational resources that will remain unrealistic for decades yet.

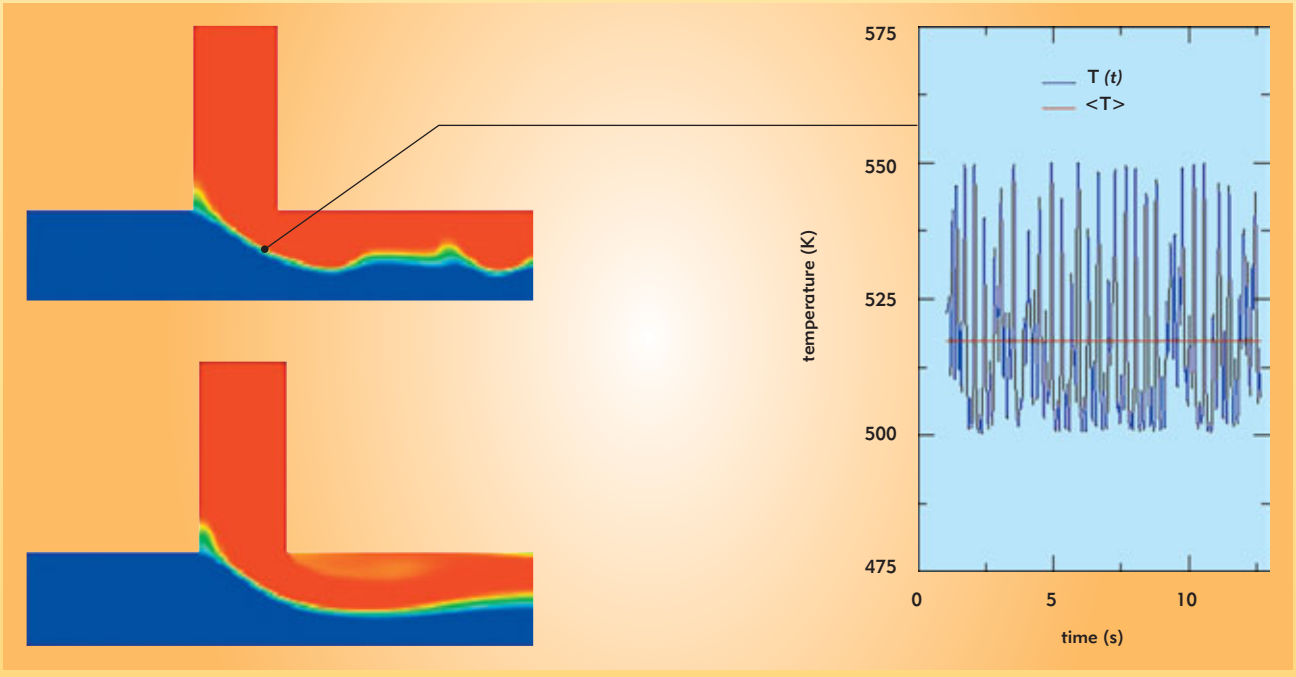


Figure. Instantaneous (top) and averaged (bottom) temperature field in a mixing situation. The curve shows the history of temperature at one point: fluctuating instantaneous value in blue and mean in red (according to Alexandre Chatelain, doctoral dissertation) (DEN/DTP/SMTH/LDTA).

The sole option, based on the fluctuating character of the flow due to turbulent agitation, must thus be to define and use average values. One of the most widely adopted approaches consists in looking at the problem from a statistical angle. The mean overall values for velocity, pressure, temperature… whose distribution characterizes the turbulent flow, are defined as the principal variables of the flow one then seeks to qualify relative to those mean values. This leads to a decomposition of the motion (the so-called Reynolds decomposition) into mean and fluctuating fields, the latter being the measure of the instantaneous local difference between each actual quantity and its mean (Figure). These fluctuations represent the turbulence and cover a major part of the Kolmogorov spectrum.(1)

This operation considerably lowers the number of degrees of liberty of the problem, making it amenable to computational treatment. It does also involve many difficulties: first, it should be noted that, precisely due to the non-linearity of the equations of motion, any average process leads to new, unknown terms that must be estimated. By closing the door on complete, deterministic description of the phenomenon, we open one to modeling, i.e. to the representation of the effects of turbulence on mean variables.

Many advances have been made since the early models (Prandtl, 1925). Modeling schemas have moved unabated towards greater complexity, grounded on the generally verified fact that any new extension allows the previously gained properties to be preserved. It should also be noted that, even if many new developments are emphasizing anew the need to treat flows by respecting their *non-stationary* character, the most popular modeling techniques were developed in the context of *stationary* flows, for which, consequently, only a representation of the flow's temporal mean can be achieved: in the final mathematical model, the effects of turbulence thus stem wholly from the modeling process.

It is equally remarkable that, despite extensive work, no modeling has yet been capable of accounting for all of the processes influencing turbulence or influenced by it (transition, non-stationarity, stratification, compression, etc.). Which, for the time being, would seem to preclude statistical modeling from entertaining any ambitions of universality.

Despite these limitations, most of the common statistical modeling techniques are now available in commercial codes and industrial tools. One cannot claim that they enable predictive computations in every situation. They are of varying accuracy, yielding useful results for the engineer in controlled, favorable situations (prediction of drag to an accuracy of 5–10%, sometimes better, for some profiles), but sometimes inaccurate in situations that subsequently turn out to lie outside the model's domain of validity. Any controlled use of modeling is based, therefore, on a qualification specific to the type of flow to be processed. Alternative modeling techniques, meeting the requirement for greater accuracy across broader ranges of space and time scales, and therefore based on a "mean" operator of a different nature, are currently being developed and represent new ways forward.

The landscape of turbulence modeling today is highly complex, and the unification of viewpoints and of the various modeling concepts remains a challenge. The tempting goal of modeling with universal validity thus remains out of order. Actual implementation proceeds, in most cases, from compromises, guided as a rule by the engineer's know-how.

(1) One may mention the spectral distribution of turbulent kinetic energy known as the "Kolmogorov spectrum," which illustrates very simply the hierarchy of scales, from large, energy-carrying scales to ever smaller, less energetic scales.
(2) This range results from the non-linearities of the equations of motion, giving rise to a broad range of spatial and temporal scales. This range is an increasing function of the Reynolds number, Re, which is a measure of the inertial force to viscous force ratio.
(3) The hypothesis that complete resolution of the Navier–Stokes equations allows simulation of turbulence is generally accepted to be true, at any rate for the range of shock-free flows.
(4) This is a problem governed by initial and boundary conditions.

Frédéric Ducros
Nuclear Energy Division
CEA Grenoble Center