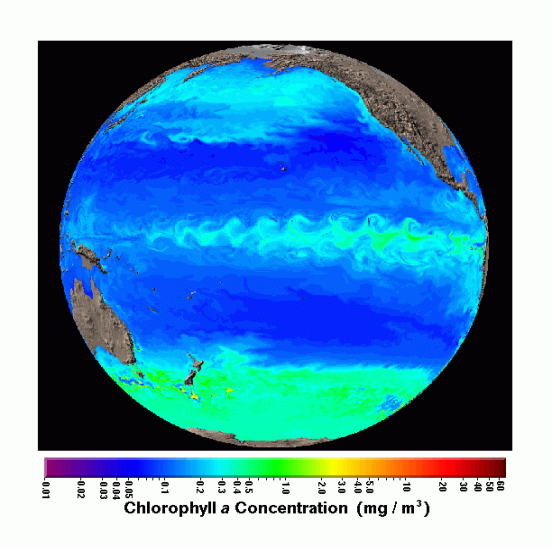
SciDAC PROGRESS REPORT

Collaborative Design and Development of the Community Climate System Model for Terascale Computing



Department of Energy

Office of Biological and Environmental Research

Scientific Discovery through Advanced Computing

PIs: R. Malone[4], J. Drake[5], **Site-Contacts:** C. Ding[2], S. Ghan[6], P. Cameron-Smith[3], J. Taylor[1], W. Collins[7], W. Washington[7], S.-J. Lin[8], **Co-PIs:** J. Baumgardner[4], T. Bettge[7], L. Buja[7], S. Chu[4], T. Craig[7], P. Duffy[3], J. Dukowicz[4], S. Elliot[4], D. Erickson[5], M. Ham[5], Y. He[2], F. Hoffman[5], E. Hunke[4], R. Jacob[1], P. Jones[4], J. Larson[1], J. Lamarque[7], W. Lipscomb[4], M. Maltrud[4], D. McKenna[7], A. Mirin[3], W. Putman[8], W. Sawyer[8], J. Schramm[7], T. Shippert[6], R. Smith[4], P. Worley[5], W. Yang[2]

[1]Argonne National Laboratory, [2]Lawrence Berkeley National Laboratory, [3]Lawrence Livermore National Laboratory, [4]Los Alamos National Laboratory, [5]Oak Ridge National Laboratory, [6]Pacific Northwest National Laboratory, [7]National Center for Atmospheric Research, [8]NASA-Goddard Space Flight Center

October 2003

CONTENTS

Page

1.0 PREFACE 5

2.0 OVERVIEW 6

3.0 SOFTWARE ENGINEERING (TOPIC COORDINATOR: PHIL JONES) 10

3.1 PERFORMANCE PORTABILITY 12
Data decomposition/data structure changes 12
Vectorization 13
Communication 14
Extensibility and Modularity 14
3.2 SOFTWARE ENGINEERING PROCESS 14
3.3 COLLABORATIVE SOFTWARE DEVELOPMENT 15
3.4 SOFTWARE FRAMEWORKS 15

4.0 COUPLER DEVELOPMENT (TOPIC COORDINATOR: J. LARSON) 17

4.1 CPL6 (T. CRAIG, B. KAUFFMAN, T. BETTGE) 20
4.2 MCT (J. LARSON, R. JACOB, E. ONG) 21
4.3 MPH (Y. HE AND C. DING) 22
4.4 INCORPORATING MLP INTO CCSM (JIM TAFT) 23

5.0 COMMUNITY ATMOSPHERE MODEL (TOPIC COORDINATOR: J. DRAKE) 26

5.1 LIN-ROOD OPTIMIZATION (A. MIRIN) 26
5.2 SPECTRAL ATMOSPHERIC DYNAMICS, PHYSICS CHUNKING AND LOAD BALANCING (P. WORLEY AND J. DRAKE) 32
5.3 PERFORMANCE STUDY OF HIGH-RESOLUTION CAM (W. YANG, M. WEHNER AND C. DING) 37
5.4 SUBGRID OROGRAPHY SCHEME (S. GHAN AND T. SHIPPERT) 39
5.5 HIGH-RESOLUTION SIMULATIONS WITH CAM2 (P. DUFFY, J COQUARD, B. GOVINDASAMY) 40
Evaluation of the multi-year time-averaged climate 41
Simulated response to 1997-1998 El Nino 43
5.6 PARALLEL I/O FOR CAM (W. YANG AND C. DING) 50
Multiple Precisions for History Buffers 50
Parallel I/O with ZioLib 50
Parallel NetCDF Libraries 52

6.0 OCEAN MODELING (TOPIC COORDINATOR: P. JONES) 53

6.1 PARALLEL OCEAN PROGRAM: POP 2.0 53
6.2 HYBRID VERTICAL COORDINATE POP 54
6.3 GENERIC GRID GENERATOR (W. VOIT, J. DAVIS, R. SMITH) 57

7.0 SEA ICE MODELING (TOPIC COORDINATOR: W. LIPSCOMB) 59

7.1 SOFTWARE DESIGN AND DOCUMENTATION 60
7.2 PERFORMANCE OPTIMIZATION 60
7.3 DYNAMICAL CORE 61
7.4 IMPROVED PARAMETERIZATIONS 61

8.0 LAND SURFACE MODEL AND RIVER TRANSPORT MODEL (TOPIC COORDINATOR: F. HOFFMAN) 63

8.1 CACHE-FRIENDLY DECOMPOSITION AND LOAD BALANCING 64
8.2 NEW ATMOSPHERE/LAND COUPLING INTERFACE 66
8.3 SUB-GRID SCALE OROGRAPHY AND RUNOFF 68
8.4 ADAPTATION TO NEW HARDWARE 68
8.5 ANALYSIS OF RUNOFF IN CCSM2 CONTROL SIMULATION 69

9.0 ATMOSPHERIC CHEMISTRY (TOPIC COORDINATOR: P. CAMERON-SMITH) 70

9.1 CCSM DEVELOPMENT (LAMARQUE, WALTERS, KINNISON, MCKENNA) 72
9.2 CHEMICAL MECHANISM DEVELOPMENT (CAMERON-SMITH, CONNELL, ROTMAN, TANNAHILL) 77
9.3 SOURCES AND SINKS OF ATMOSPHERIC METHANE (TAYLOR, CAMERON-SMITH) 80
9.4 MARINE AEROSOL AND GAS PHASE INTERACTIONS (ERICKSON, TAYLOR) 83
9.5 MODEL VALIDATION (CAMERON-SMITH, LAMARQUE, TAYLOR) 84

10.0 BIOGEOCHEMISTRY (TOPIC COORDINATOR: D. ERICKSON) 87

10.1 FINE-RESOLUTION POP SIMULATIONS (S. ELLIOT, S. CHU, M. MALTRUD) 88
10.2 IRON ENRICHMENT STUDIES (M. MULTRUD) 89
10.3 GLOBAL DMS FLUX FROM THE OCEAN USING POP (SHAOPING CHU, MAT MULTRUD, SCOTT ELLIOTT, JOSE HERNANDEZ, DAVID ERICKSON) 90
10.4 BIOGEOCHEMISTRY DIAGNOSTICS (K. CALDEIRA, S. DONEY) 92

11.0 COORDINATION AND MANAGEMENT (R. MALONE AND J. DRAKE) 95

11.1 MANAGEMENT PLAN 95
11.2 COORDINATION WITH SCIDAC INTEGRATED SOFTWARE INFRASTRUCTURE CENTERS AND COLLABORATORIES 97
11.3 COORDINATION WITH OTHER DOE/SCIDAC CLIMATE MODELING PROGRAMS 98
11.4 CURRENT TASKS AND LEVELS OF EFFORT 98

PUBLICATIONS DURING 2001-2003 102

12 APPENDIX A: MANAGEMENT PLAN 105

12.1 INTRODUCTION 105
12.2 OBJECTIVES 105
12.3 MULTIPLE LEVELS OF COORDINATION 105
Principal Investigators 106
Laboratory Coordinators 106
Topic Leaders 107
Topic Coordinators 107
Working Groups 107
Scientific Steering Committee 108
CCSM Advisory Board 109

13 APPENDIX B: NAMES, AFFILIATIONS, AND PRIMARY INTERESTS 111

1.0 PREFACE

In this, our third progress report, we examine our progress from the perspective of our original proposal to the Scientific Discovery through Advanced Computing (SciDAC) Program. We attempt to evaluate whether our original objectives were appropriate; to what extent they have evolved and why; what progress has been made toward those objectives; and evaluate the benefit of the "SciDAC Consortium" to the National Center for Atmospheric Research (NCAR) Community Climate System Model (CCSM), the focal point of this entire project. We also examine various aspects of the management of such a large cooperative program such as the degree of cooperation among the six U.S. Department of Energy national laboratories, one National Aeronautics and Space Administration Office, and NCAR.

Such a retrospective view is appropriate at the half way point in this five-year project. Also, the project will be undergoing a review at Oak Ridge National Laboratory in October 2003, and we hope this progress report will be useful to the review committee members. To address this second purpose, we have included some material from previous reports, in addition to new progress. In particular, we have included a copy of the Management Plan in the Appendix.

Chapters are devoted to each of the major components of the CCSM Consortium Project, software design and engineering, the "Flux Coupler" that links the components, the component models themselves, and two scientific development activities: the collaborative development, installation and testing of packages for atmospheric chemistry and ocean biogeochemistry. In each chapter, we go back and reexamine the original objectives and note changes that have been made. We have attempted to give each chapter a uniform structure and to address a common set of questions in each section. Each chapter of the report is organized into the four sections given below.

- *Background*
- *Objectives*
    - What were the original (proposed) objectives and priorities?
    - Have the objectives been modified substantially? If so, why?
    - How well do the project objectives align with NCAR and CCSM objectives?
- *Progress*
    - What progress has been made toward accomplishing those objectives?
    - Is this rate of progress adequate to meet the objectives within the 5 year period? If not, why not?
    - What objectives have been completed?
- *Plans*
    - For the remainder of the project, what are the priorities?

A different set of questions is addressed in the final section on management.

2.0 OVERVIEW

This report marks the half way point of the Community Climate System Modeling (CCSM) Consortium Project sponsored by the U.S. Department of Energy (DOE) Office of Biological and Environmental Research, Scientific Discovery through Advanced Computing (SciDAC) Program. It shows rapid progress on several tasks within the proposal. The teams addressing individual tasks are now well-established and working in consort. Six DOE national laboratories (Argonne National Laboratory {ANL}, Lawrence Berkeley National Laboratory {LBNL}, Lawrence Livermore National Laboratory {LLNL}, Los Alamos National Laboratory {LANL}, Oak Ridge National Laboratory {ORNL}, and Pacific Northwest National Laboratory {PNNL}), the National Aeronautics and Space Administration (NASA), Data Assimilation Office (DAO), and National Center for Atmospheric Research (NCAR) are actively involved. The degree of cooperation among these organizations is exceptional and is a legacy of the cooperative spirit instilled in previous DOE climate projects by David Bader, former DOE program manager of Computer Hardware, Advanced Mathematics and Model Physics (CHAMMP), Climate Change Prediction Program (CCPP), (see the section on "Relationship to other programs"), and SciDAC climate programs.

Software engineering is a key focus of this project as we work cooperatively with NCAR to provide the modeling framework and culture that allows the model to be easily extended while maintaining a high level of parallel performance on multiple computing platforms. The CCSM Software Engineering Working Group (SEWG) was formed at the onset of this project and SciDAC Consortium members have contributed well to the activities of the group. Increases in simulation throughput are directly attributable to this project for the atmosphere, land, ocean and sea ice components. A new version of the CCSM coupler is also a result of this project, built on a utility, toolkit layer produced by this project. While our focus has been on performance of working code we have worked closely with the NASA Earth System Modeling Framework (ESMF) project and the SciDAC Common Component Architecture (CCA) who are effectively defining the software infrastructure for future model developments.

As the project has modified its software engineering tasks to adapt to the changing supercomputing environments, it has also directed effort into new areas that extend the CCSM simulation capabilities and develop the component models. The development of a new ocean model with a hybrid vertical coordinate, (HYPOP), has made significant progress. The inclusion of atmospheric chemistry in the Community Atmosphere Model (CAM) is now showing results with good comparisons to observational data. Finally, simulations with ocean biogeochemistry are extending the scientific scope of the model and will lead to the simulation of chemical cycles (e.g., carbon) in a coupled earth system model.

The first steps toward a CCSM model following an "open" design have now been succeeded by the second and third steps with the release of the CCSM2 in May 2002 and the establishment of a Change Review Board for the atmospheric model. A new model is nearing completion that will address the requirements for the Intergovernmental Panel on Climate Change (IPCC) Simulations and will be released to the community in the near future.

Progress in code development in accordance with the proposed designs of the atmosphere and coupler components has resulted in significant performance improvements to the CCSM2. A new version of the community land model (CLM) 2.1includes new data structures and optimized parallel algorithms. The incorporation of the projects coupler utilities, MPH and Modeling Coupling Toolkit (MCT) are evident in the new coupler, CPL6, which has been adopted in

CCSM and was included in the last CCSM release. Two utility libraries that were released, MCT1.0 and ZioLib, show a new level of code maturity and robust performance.

Other significant progress discussed in this report are the optimization of the 2D decomposition of the Lin-Rood (finite volume {FV}) dynamical core, the introduction of new parallel spectral decompositions in the Eulerian Spectral dynamical core and scalability improvements from load-balanced parallel decompositions of the "chunked" physics. The load balancing will become even more important as the model is exercised with atmospheric chemistry options.

A new version of Parallel Ocean Program (POP), version 2.0, has been completed and was released in April 2003. It differs from POP 1.4.3, the last formal release, in a number of ways. Partial bottom cells and the new tripole grid have been implemented in POP2.0 expanding the model options. But foremost, it is designed to have either very small computational domains that can be contained in cache or much larger domains that are compatible with vectorization. This allows POP to perform well on vector machines like the Japanese Earth Simulator (ES) and the new Cray X1, while remaining compatible with cache-based machines.

An important step took place in the development of the new hybrid vertical-coordinate version of POP, called HYPOP. One of the design requirements for HYPOP is that the barotropic mode be solved by explicit subcycling, rather than by an iterative implicit technique as in POP. Several attempts at splitting the barotropic (2D vertically averaged velocity) and baroclinic (3D velocity deviation from the barotropic) proved to be unstable. A new method has been developed that is not only stable but is capable of retaining the high-frequency, high-wave number, gravity-wave modes of the barotropic velocity. Other methods must damp these modes to prevent numerical instability. The method has been implemented and tested in HYPOP.

The dynamic cores of the LANL sea-ice model (CICE), and Community Sea Ice Model (CSIM), the sea ice component of CCSM, have been improved with the addition of metric terms in the momentum equations and with an incremental remapping scheme for ice transport. The remapping scheme is more than twice as efficient as the previous MPDATA scheme and also is more robust numerically. In recent months CICE has been rewritten to be more vector-friendly; vector performance has improved by a factor of eight on the Fujitsu VPP5000. CICE in now being tested on the ES and Cray X1, and the vector changes are being added to CSIM.

Incorporation of tropospheric chemistry moved from the design phase to the implementation phase with multi-year global simulations and the first diagnostic look at greenhouse gases advected by the model. A configuration of the model using the FV core and the Whole Atmosphere Community Climate Model (WACCM) stratospheric chemistry package was also tested for performance and portability.

The organizational and technical aspects of the biogeochemistry task were defined. Representatives from ORNL, LANL, LLNL, and ANL held discussions about how to conjoin their efforts, and a draft plan was produced. Ocean simulations with biogeochemistry were made to study nutrient upwelling and chlorophyll production, to study the effects of iron enrichment, and to compute the worldwide distribution of dimethyl sulfide (DMS). Completing the link of sulfur cycle by driving an atmospheric chemistry simulation with the computed flux from the ocean and finally producing a fully coupled simulation is a goal for the next part of the project. Merging the Consortium efforts with other Biogeochemistry Working Group projects and supporting the rapid incorporation of these processes in the CCSM is a particular focus of the coming year.

The Project continued to hold weekly teleconferences and occasional Access Grid meetings. Interactions with closely related SciDAC projects, the SciDAC Performance Evaluation Research Center (PERC), and the Earth System Grid (ESG) Collaboratory, continued with ongoing projects. Joint meetings with the CCA software developers were quite helpful in terms of aligning the needs and plans of the Consortium with the plans and products of the SciDAC Integrated Software Infrastructure Centers (ISICs) and Collaboratories.

Because the Consortium Project is specifically focused on the CCSM, nearly all Consortium participants were present at the annual CCSM summer workshop in Breckenridge, Colorado at the end of June 2003. On the day prior to the beginning of the workshop, we convened a one-day meeting of Consortium participants to review progress and make plans for the next six months. This meeting has become an annual event because it allows us to discuss our own efforts in the context of the entire CCSM community's activities.

Climate research and the CCSM Consortium Project were highly visible at SuperComputing 2002 and are active in the planning for SuperComputing 2003. All of the climate-related SciDAC projects mentioned in the preceding paragraphs cooperated in putting together an impressive display in the SciDAC booth.

The Management Plan, (included as a Milestone/Deliverable in the last progress report), was followed by the Memorandum of Understanding (MOU) with the NASA ESMF and was put into practice. Representatives from our project are active participants in ESMF meetings, design review and code development. Software designs are taking ESMF and CCA component structures into account and prototype implementations of selected modules is underway. A meeting with the SciDAC CCA project and ESMF representative resulted in the definition of tasks for the MCT utility and for the atmospheric model software development. The CCA hosted two "climate camps" in which Consortium members participated.

The following milestones have been attained, resulting in the indicated deliverables. The third column shows the section in this report where the relevant work is discussed in more detail.

TABLE 1. MILESTONES AND DELIVERABLES SUMMARY

Milestone	Deliverable	Section
Expanded version of MCT1.0	MCT software release	4.2
MPH3.0 available	MPH software release	4.3
Incorporation of new coupler CPL6	Verified code	4.1
2-D decomposition of DAO finite-volume dycore	Verified code in CAM2.0	5.1
Scaling of spectral dynamics, Physics chunking and load balancing	Verified code in CAM2.0.2	5.2
Load balancing of physics chunks	Verified code in CAM2.0.2	5.2
Vector-friendly version of CAM2	Preliminary implementation	3.1
High resolution study of CAM2	Publication	5.5
Subgrid precipitation parameterization with CAM2	Simulation (publication)	5.4

Requirements document for POP	Draft document	3.2
Cache-friendly chunking in POP2.0	Beta release of POP2.0	3.1
Vector-friendly version of POP	Implementation of POP1.4.3	3.1
MLP implemented in CCSM2	Performance study	4.4
HYPOP tested in Lagrangian and Eulerian limits	Validation study	6.2
ZioLib software release	Software release	5.5
Parallel I/O (NetCDF) in CAM2	Verified code in CAM2.0.2	5.5
Requirements document for land model	Revised and accepted	8.0
CLM2.1/CAM interface improvements	Verified code in CAM2.0.2	8.2
Vector-friendly CLM2 (rewrite)	Preliminary version	8.4
Small tropospheric Chemistry implemented	Simulation (publication)	9.2
Ocean biogeochemistry implemented in POP	Simulation (publication)	10.1
ESMF collaboration	Amended design document and prototype code	11.2
PERC collaboration	Performance feedback	11.2
ESG collaboration	Supercomputing Demo	11.2
CCA collaboration	Prototype code	11.2
SDM collaboration	Grid ftp, NetCDF and HRMs	11.2
TSTT collaboration	Grid generation	11.2

3.0 SOFTWARE ENGINEERING (TOPIC COORDINATOR: PHIL JONES)

Background

Software engineering is typically defined as a formal process used to design and implement software. In this document, we will broaden the scope of software engineering to include several other issues that are typically classified as software issues, namely collaborative software development, use of software frameworks and the specific design goals of performance portability, modularity and extensibility. With this broad scope, software engineering underlies much of the effort in this project.

Performance portability is the ability to achieve good performance across a variety of computer architectures while maintaining as close to a single source code as possible. Such a design goal was chosen as a prime focus of this project and even appears in the title. As a community model, CCSM must run on a variety of machine architectures available to the climate community and must perform well on all these architectures for the computationally intensive climate simulations.

Similarly, a community research model like the CCSM must be modular and extensible to enable rapid adoption of new capabilities and new physical parameterizations. Modularity also permits users to choose between many model configurations to customize the model for specific applications.

A formal software engineering process involves a cycle of defining requirements, designing the software, implementing the design and testing the software. Documentation and reviews at each stage are important to catch bugs early and to avoid costly rewriting when designs do not satisfy current or future requirements. Testing is performed at several levels. Unit testing of individual subroutines and modules verifies modules work as designed; integrated testing of entire models can be used to validate physical fidelity of the model and ensures that all components of a model interact with each other without unintended side effects. Frequent regression tests are used to catch problems generated due to changes in computational environment or bugs introduced during minor maintenance. Following such a formal process has been shown to reduce development time and improve the robustness of resulting software.

Finally, because this project involves a large number of distributed developers, tools and processes are required for managing code changes and conflicts. Such tools include common software repositories and procedures for introducing code into the repositories. Assignment of responsibilities for testing, quality assurance, conflict resolution and scheduling of changes are also necessary for a large collaboration.

Objectives

<u>Performance Portability</u>

As mentioned in the previous section, performance portability is a prime objective of this project and is also an important CCSM objective. At the start of this project, the target machines in the climate community were computers built from clusters of commodity cache-based microprocessors. More recently, vector computers have become available to the climate community and have added a significant new challenge to performance portability.

There are cases where portability is not possible within a single source code. In these cases, we hope to isolate non-portable code into modules or libraries that can be selected at compile time or at run time. Generic interfaces or wrappers can be defined so that the calling code can remain portable. An example of such a structure is the support of different communications paradigms. Often communication-related routines can be isolated in a small set of routines and called using an interface that is identical whether the underlying code is implemented in Message Passing Interface (MPI), SHMEM, Co-array Fortran or copies to a memory buffer. Another goal of our effort is to avoid use of preprocessor directives (e.g., the C preprocessor ifdef) as a primary means of achieving portability; directives often can proliferate and adversely affect code readability. Selective use of directives is permitted in cases where code is hidden from the user or well encapsulated.

Climate codes tend to be limited in performance by memory bandwidth so a key aspect of performance portability is the need to adjust the size and shape of data structures to optimize performance on machines with different cache sizes or with vector processors. A goal of this project is to provide tunable parameters for adjusting data structure size and loop lengths to optimize codes based on system size, problem size and processor configuration. Automated adjustment of these parameters is a long-term goal, particularly when used to achieve dynamic load balancing. In the short-term, parameters should be sufficiently easy for scientists to adjust without detailed knowledge of memory bandwidth properties of a particular architecture.

Extensibility and Modularity

Modularity and extensibility are desirable goals at all levels in the CCSM model. At the lowest level, the ability to swap or choose between physical parameterizations is necessary and was already designed into most CCSM component models. Component models can also be modular at the level of major model subcomponents. For example, it is often useful to treat atmosphere model dynamics and physics as self-contained models. Finally, at the model coupling level, it should be possible for alternative component models to be substituted or additional physical components (e.g., biogeochemistry) to be added. Since physical parameterizations are not the focus of this project, our goals have mainly been on the higher two levels. In particular, at the start of the project the atmosphere model did not have a clean separation between the model physics and the dynamical core, making the introduction and use of different dynamical cores problematic. At the highest level, flexibility in defining components and fields is a goal in the development of a new coupler for CCSM.

Software Engineering Process

As described above, a formal software engineering process includes: defining software requirements, creating a detailed design, coding to the design and testing the resulting software. Because this effort is primarily dealing with legacy code, the process must be adapted to fit the current software. Adopting the entire process is a goal for all new code developed under this project. For existing code, the goal is focused primarily on testing and validation.

Collaborative Software Development

Because of the large number and geographical distribution of collaborators in this project, an early goal was to develop the procedures and infrastructure necessary for all collaborators to contribute effectively. In particular, it is necessary for contributions to be adopted as soon as they are ready, yet not interfere with other model development efforts. This requires a high level of integration with access to a common repository. It also requires procedures for testing and

committing code so that changes will not introduce bugs or otherwise adversely affect the model code.

Software Frameworks

One means of achieving portability and consistency between models and reducing the workload associated with code maintenance is to utilize software frameworks and libraries to provide common functionality. Several framework efforts are currently in progress and are being tracked by this effort. The NASA-funded ESMF in particular is being developed specifically for the climate, ocean and weather community and has many joint investigators within the CCSM effort. A goal of this project is to track these efforts and adopt such frameworks when feasible.

Progress

3.1 PERFORMANCE PORTABILITY

Data decomposition/data structure changes

Performance in climate models is strongly constrained by memory bandwidth and the key to performance portability is the ability to tune the data structures and corresponding data access for a given platform. We have adopted schemes in both the atmospheric physics and ocean models that rely on "chunks" or blocks of data that can be tuned for a particular architecture. The chunks or blocks can be sized for either cache or vector machines, distributed in a load-balanced manner and oversubscribed to processors to provide hybrid thread/messaging-passing programming models.

In the atmosphere model, the physics data structure is now broken up into chunks of (logically) contiguous data points. This required removing the old latitude slab decomposition in favor of a chunk or set of chunks whose size is determined at run time. To achieve load balancing necessary for optimal performance, the chunks can be distributed across processors and threads using a number of different static load balancing schemes, three of which are part of the official release. The schemes vary from load balancing within the same domain decomposition used by the dynamics (requiring no interprocessor communication) to an almost exact load balance that requires additional interprocessor communication. The new data structure includes a compile time specification of the maximum size of the basic data structure (for serial performance efficiency) as well as a runtime determination of the actual size, to support the runtime specification of processor counts, MPI tasks, and OpenMP threads per MPI task. The size parameter can also be used to improve vectorization or cache locality, and to expose additional MPI or OpenMP parallelism. This new decomposition scheme required new routines to redistribute data between the atmospheric dynamics and physics and also to prepare data for flux coupler when run in coupled mode. We continue to maintain the physics chunking, helping NCAR implement the data structure in new physics routines (e.g., sulfur cycle, ozone, and other aerosols).

The spectral dynamical core was also redesigned, replacing the 1D parallel decomposition used in the Fourier and spectral spaces. Blocking the spectral transforms improved serial performance, decreased interprocessor communication, improved load balance, and decreased memory requirements.

Changes to the ocean model were similar to the chunking in the atmospheric physics. The ocean model grid is divided into a number of 2D Cartesian blocks which can be distributed across

processors in way which either optimizes load balance (for the computationally-intensive baroclinic modes) or optimizes communication (for the communication-dominated barotropic mode). Data redistribution between the barotropic and baroclinic sections is performed if necessary. In the ocean model, an added advantage of the blocking scheme is that some blocks are completely land and are eliminated from the domain. For performance reasons, the block size is determined at compile time but the number of blocks per processor is assigned at run time based on the run-time specification of processor configuration. Smaller block sizes are best for cache-based machines and for maximum land point elimination and load balancing. Large block sizes are used on vector architectures. The new blocking scheme has shown up to 30% performance improvement for high-resolution simulations through both cache improvements and land point elimination. At coarse resolution, there are few blocks that are completely land and the block size was already cache-friendly, so little improvement was seen. However, there was no penalty at coarse grid sizes when the blocks were configured identically to older versions of the model.

Vectorization

Vector computers have once again become available to the climate community with access to both the Cray X1 and the Japanese ES (through collaboration with the Central Research Institute for Electric Power Industry, {CRIEPI}). Progress has been made to expand performance portability to include vector architectures.

The ocean model ported easily to vector architectures as it uses array syntax over horizontal domains, an easily vectorizable construct. However, there were a few routines written for improved cache performance that had to be revised for vector architectures. The resulting performance is shown in Figure 1.

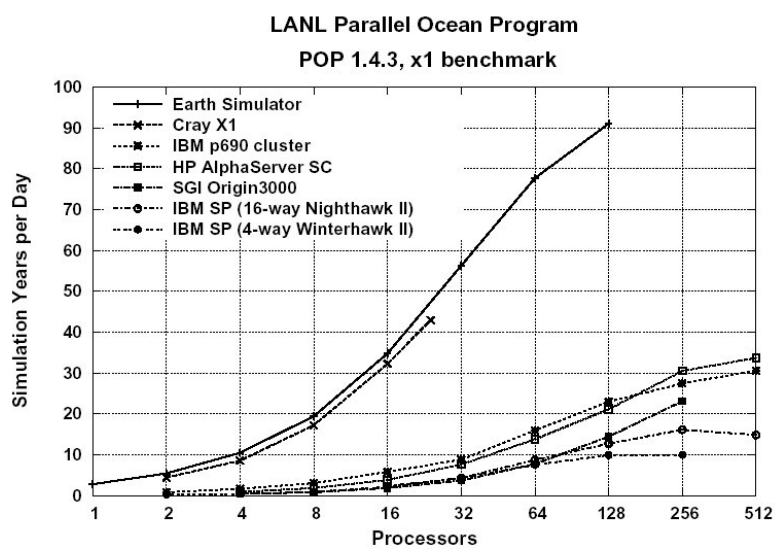


Fig. 1

The CICE model performed poorly on vector architectures due primarily to the structure of the ice thermodynamics. This part of the sea ice code contained outer loops over the horizontal domain with subroutine calls and branching within the loop. In collaboration with CRIEPI (through Clifford Chen of Fujitsu), the ice thermodynamics was restructured, pushing the horizontal loops into subroutines and replacing conditionals within loops with a pre-gather construct. The resulting code vectorizes well and also performs well on cache-based architectures.

Vectorization of the atmosphere and land models is in progress. In the atmosphere, the chunk size parameter is used to maximize vector lengths in the physics, but the structure of some physics algorithms are preventing vectorization and will need to be revised. Local loop rearrangements in the spectral dynamical core are exposing more vectorization to the compilers. The land model structure was not vectorizable and much work will be required to design a new structure for the land model.

Communication

Progress has been made to eliminate some performance bottlenecks related to communication. Communication between atmosphere and land when running without the coupler was performed using a gather-scatter through a single process. This has been replaced by a more general all-to-all communication pattern. Other work includes a number of interprocessor communication optimizations, both within the atmospheric dynamical core and between the physics and the dynamical core. Some of these are system-specific, and the intent is making these runtime options.

Use of alternative messaging paradigms is also being explored, particularly to improve the scaling of the ocean model's barotropic solver. Performance and scaling of this iterative solver depends strongly on message latency, so low latency messaging results in large performance improvements in the model. On the Cray X1, Co-array Fortran is supported as a low-latency alternative and the use of Co-array Fortran for the communications in the barotropic solver resulted in improved scaling on the X1. On machines with shared memory, the Multi-Level Parallelism (MLP), developed by Jim Taft, permits the use of shared memory buffers for messaging, providing message passing at the cost of a memory copy. Early results showed dramatic improvement both in POP and in the CCSM coupler using MLP, but performance has not yet been demonstrated in production versions of these codes.

Extensibility and Modularity

Improving modularity in this project has focused mainly on the support of multiple dynamical cores in the atmosphere model. A design and implementation for the separation of physics and dynamics in CAM was achieved, allowing the inclusion of multiple dynamical cores as well as the new physics data structure and parallel decomposition described previously. This progress was critical in the evaluation and testing of the new FV dynamical core and for the continued optimizations of physics and dynamics.

3.2 SOFTWARE ENGINEERING PROCESS

A goal of this project was to follow a more formal software engineering process. Early in the project, requirements documents were written for all the components, but detailed design documents for all models were never completed. There has been recognition in this project that these early stages of the software engineering process are not as useful when working with codes that are already well developed. For new modules and for the new hybrid-coordinate ocean code, some attempts at following this process are being made.

Other aspects of the software engineering process are more directly relevant to this effort and are being used effectively. A more rigorous process for defining code releases and for committing codes to the repository is being used. Testing of all the models has also become more frequent and a series of automated tests have been developed, particularly for the atmosphere model. Full

documentation is now available for all models; all component codes now include self-documenting interfaces in which the proTeX package can extract and build interface documents directly from the source code, enabling up-to-date and consistent documentation.

3.3 COLLABORATIVE SOFTWARE DEVELOPMENT

Because of the large number of collaborators in this project, early efforts were made to insure code improvements could be rapidly implemented and shared with other developers. In practice, this required a common software repository with procedures and requirements for committing code to the repository. Change review boards to manage this process have been implemented in some form for all models, though the atmosphere model requires the most formal process. The process has been successful and has further evolved to include prioritizing and scheduling of commits to the repository and web-based tools to make this information readily available to all developers.

3.4 SOFTWARE FRAMEWORKS

Software frameworks provide a means of achieving portability and can reduce workload by providing a well-tested utility layer that can be shared across all components of a coupled model. Framework code also ensures consistency across the applications when shared functions are used. The NASA-funded ESMF is an effort to supply utilities and coupling tools to the climate, weather and ocean communities. ESMF will provide common time management and calendar utilities, error and logging functions and some performance monitoring interfaces. Utilities for defining grids, regridding or interpolation and communications for data reorganization and halo updates will also be included in ESMF. Finally, ESMF will provide fields, field bundles and components for use in coupling model components together.

Version 1.0 of ESMF was released in May 2003 with some limited functionality, including ability to couple simple models together. Utilities like the time manager were also included in this release. Monthly intermediate releases are being made available to ESMF friendly users and developers (including CCSM developers). Some work has already begun to create a prototype CCSM coupler using ESMF libraries and enough functionality should be available in the next year to begin adopting ESMF more fully throughout the CCSM.

Plans

Plans for the future of the software engineering component of this project will largely continue along the lines above. Performance will be a high priority. Continued performance evaluation in collaboration with PERC will identify and fix performance bottlenecks using profiling and interfaces to hardware counters. Work on data structures will continue, with particular emphasis on efforts to make the land model structure friendlier to vector architectures. As the project moves toward higher resolution models, performance and memory issues at such high resolutions will need to be addressed. New communication paradigms for parallel architectures will continue to be evaluated.

As new software is developed, particularly for the hybrid-coordinate ocean model and for biogeochemistry models, formal designs must be generated to speed the development process. New test suites for components other than the atmosphere must be created and used regularly to reduce bugs and ensure correctness. Use of ESMF code when available will allow CCSM developers to concentrate on details of the model code rather than other computational

infrastructure. These software engineering efforts will continue to support all parts of this project without impeding progress necessary to meet programmatic goals.

4.0 COUPLER DEVELOPMENT (TOPIC COORDINATOR: J. LARSON)

The coupler plays a central role in making the separately developed component models (atmosphere, ocean, etc.) function as a combined system. It *controls* information exchange between components and *synchronizes* the time marching of components. Besides complex synchronization issues and physical quantity (such as heat fluxes) computations, the information exchange requires substantial software development (MCT), and appropriate mechanisms of integrating individual components into a single system be conveniently handled (MPH).

Currently, CCSM is using a new coupler, CPL6, which is a joint creation of NCAR and SciDAC scientists. CPL6 makes extensive use of two software libraries developed under this project, MCT and MPH. CPL6 meets or exceeds the performance of its predecessors and represents a major advance in software design and implementation compared to previous NCAR couplers and would not have been possible without the work of SciDAC scientists. MCT and MPH are in turn each available as standalone software products and have been used in other projects.

Background

At the beginning of this project, CCSM's coupled integration scheme had been in place for several model versions going back to 1996. First, each component is a separately compiled executable. There are five executables: the atmosphere, ocean, sea ice and land models and the coupler. All models exchange data only with the coupler. The time integration scheme is to allow each model to run concurrently, to the extent that inter-model data dependencies allow. The models exchange data in a predetermined order that is encoded into the coupler's main routine. The exact data exchanged is also hard-coded. The atmosphere, sea ice and land models exchange data every simulated hour and the ocean communicates once per simulated day. In addition to controlling the data flow and model integration, the CCSM coupler also interpolates data between model grids and performs one important piece of model physics: calculating the ocean-atmosphere fluxes of heat and momentum. The interpolation is performed as a matrix-vector multiply. The elements of the matrix were calculated offline using a program from LANL called SCRIP. The coupler also performs diagnostics on the total fluxes of heat, water and momentum and writes its own history and checkpoint (restart) files.

CPL5 was the version of CCSM's coupler in use internally at NCAR at the start of the project. As NCAR switched its computing platforms from vector to microprocessor based parallel computers, each component model was parallelized using MPI except for the coupler which only had shared memory parallelism using OpenMP. This was seen as a potential bottleneck to achieving portable high performance on distributed memory computers. In addition, whenever a new physical variable had to be added to the coupled system and passed between two or more models, a complex set of changes had to be made to several areas of code. These changes were not well documented and, if done incorrectly, could create errors that were difficult to track.

The ACPI Avante Garde project (AG) preceded the current SciDAC effort. The AG project was also concerned with CCSM development but focused only on the atmosphere and coupler. The AG effort produced two supporting software libraries created for improving the CCSM coupler. The first package of supporting software, MCT, provides datatypes and methods for building parallel couplers through a set of Fortran90 modules and two small libraries. In addition to MCT, a Fortran90 module, called the MPH was created under AG. MPH handles the critical task of helping each stand-alone component-model executable to recognize the existence of other components within CCSM and getting necessary processor information. MPH provides several possible independent model integration mechanisms. It allows component model processor

geometries to be specified in a small input file. It also provides facilities for standard out redirection and joining of MPI communicators. Towards the end of AG, both MPH and MCT were being used in prototypes of the new CCSM coupler. Two versions of MPH were released to the public during the AG project.

Objectives

The original objectives of the proposal for the coupler assumed the completion of a distributed memory coupler for CCSM built on top of MCT and MPH by the end of the AG project. Development of the coupler was to then focus mostly on performance and flexibility. For performance, OpenMP threading would be added to MCT to possibly increase performance on systems with large SMP nodes. We would also work to optimize the single processor performance and do platform dependent tuning where appropriate. Finally, collective operations within MCT were to be checked for scalability to thousands of processors.

Several proposed new features of the coupler fell under the term "flexibility." We envisioned the models within CCSM possibly performing two types of dynamic load balancing. First, the models may redistribute their grid points within each model's fixed pool of processors. Propagating that information to the coupler and updating its decomposition descriptions in an efficient manner would require changes both to MCT and the coupler itself. A second type of possible load balancing is when the models give or receive processors to each other depending on the load. Accommodating this type of load balancing would require modifications to the coupler, MCT and MPH.

Another proposed area for increased flexibility was allowing for dynamic grids such as those found in adaptive mesh refinement schemes. This would require frequent online calculation of interpolation weights possibly using an MCT interface to a parallelized version of the SCRIP program. Vertical interpolation and interpolation between 3D fields were also proposed features which would require defining new MCT data storage and decomposition description classes to handle 2D slices and 3D blocks. (At the end of AG, MCT's data storage type treated everything as a collection of 1D vectors).

We envisioned further abstracting MCT datatypes to simplify the process of creating a coupled model and to allow more flexibility in the number of total models coupled and the fields exchanged. Features such as a component model registry and standardized names for fields involved in coupling could allow many of the inter-model interactions to be determined at runtime. Finally, much of the coupler functionality was to be eventually folded into a CCSM utility layer that would be used both within and between model components.

The above objectives have been modified substantially since the beginning of the project. The most significant reason was that CPL6 was not finished at the end of the ACPI-AG project. Although MCT had enough features to provide data parallel versions of the main functionality in CPL5 and the end of AG, the last of these, the sparse matrix interpolation class, was only completed a few months before the completion of AG. Also, conferences with the NCAR coupler developers revealed that additional classes would be needed to meet all the requirements and that writing an entirely new coupler would be best. An ambitious plan was developed where a new coupler would be written entirely from scratch using CPL5 as a guide for the necessary functionality. Completing CPL6 and validating it against CPL5 became the main priority for the coupler sub-project in the first 24 months of the SciDAC climate project. The MCT team took on the additional task of helping validate, develop and port CPL6 as well as measure its performance within the fully coupled system.

The anticipated demand for dynamic load balancing has not yet appeared within CCSM and there are no major developments planned for CCSM within the remainder of the SciDAC timeframe that would require sophisticated inter- or intra-model load balancing. Thus this work will be done on an exploratory basis in the final years of the project.

The tentative plans for incorporating new models may not require the transfer of large 3D fields between models and the coupler. For the case of atmospheric chemistry, the new code will most likely be implemented as additional atmosphere model column physics. The coupled fields are still predominately seen as 2D. Thus changes to support higher dimensional data transfers and decompositions have also been deferred.

Plans for a CCSM-wide utility/machine layer have been placed on hold because it largely duplicates the substantial effort being exerted to create the ESMF, which began shortly after SciDAC. Because ESMF chose to start with a clean slate and write its lowest level classes in C++, there has been no direct use of MCT within ESMF although there are many similar concepts between the two. CCSM has plans to adopt ESMF provided it can meet performance and other requirements. Since ESMF can handle much of the functionality of CPL6, MCT and MPH, development of each of these tools in the final years of SciDAC will depend on which parts of ESMF are adopted by CCSM and when. A memo of understanding developed between ESMF and SciDAC investigators agreed that SciDAC would support the final decision of the CCSM management regarding adoption of ESMF within CCSM.

The original emphasis on performance remains an ongoing task. The revival of vector architectures such as the Cray X1 and the ES has placed new demands on performance portability within MCT. We have new high-priority tasks to incorporate vectorized versions of the central processing unit (CPU)-intensive parts of MCT within the cache-friendly source code. The performance of other climate models on the ES created a new objective to try using Multi-Level Parallelism (MLP-parallelism on platforms from SGI).

On current distributed memory computers, it remains a cumbersome task for independent component models (executables) to recognize each other. MPH handles this handshaking task. It should meet the following requirements: (a) Flexible component names. As CCSM is developed, component model and their names evolve, e.g., atmosphere model changed from (Community Climate Model (CCM) to CAM. Component names cannot be hardwired into the coupler. (b) Allow several model integration mechanisms. In CPL5, each component model is a single executable. In the related PCM (parallel climate model), each component model is a subroutine, and all component models are compiled into a single executable. As CCSM evolve, a component model could have several sub-components. Finally, ensemble simulations require yet another multi-instance mechanism. (c) Resource allocation. Processor allocation must be flexible and only need to be specified at runtime through a simple controlling mechanism. All these requirements are met by MPH. In additional, a number of further utilities are provided as well.

CPL6, which uses MCT and MPH, is already well suited to support all of the science objectives identified in the CCSM Science Plan for 2004-2008 (REF). The SciDAC coupler team has a close working relationship with the NCAR coupler developers so that the SciDAC work is always in alignment with NCAR/CCSM objectives.

New development in MPH was also required. In anticipation of future CCSM requirements, supporting ensemble integrations and multiple-instances of a sub-model within a coupled were added under SciDAC.

Progress

Progress can be organized within 3 main tasks: Progress in general towards CPL6 and specific progress on the CPL6 subsystems: MCT and MPH. We will also describe here an attempt to incorporate a different kind of parallel library, MLP, into CCSM.

4.1 CPL6 (T. CRAIG, B. KAUFFMAN, T. BETTGE)

Construction of the new coupler began shortly before the start of this project (Spring 2001) with a team led by NCAR employees Tom Bettge, Tony Craig and Brian Kaufmann and also including the ANL MCT development team. CPL6 development took about two years total. Additional higher level classes deemed necessary to meet all of NCAR's requirements for the new coupler, including streamlining the process for adding fields to the coupled system and presenting a simple application program interface (API) to the individual component models, were created. These CPL6 classes include: bundles, maps, domains, and contracts, and are all built on top of MCT classes. CPL6 was feature-complete by the Spring of 2003 when it was hooked up to all of the components of CCSM (CAM, POP, etc.) and replaced CPL5 as the coupler for the "in-house" version of CCSM at NCAR. A public release of the new CCSM with CPL6 is expected at the end of 2003. CPL6 was written completely from scratch and its completion and adoption by CCSM in two years represents a major accomplishment of the NCAR-SciDAC team. CPL6 meets the following requirements:

- Allow components to be on distinct sets of processors.
- Allow any number of total processors and processors-per-component.
- Allow other models to be added easily.
- Allow each component to have its own numerical grid and decomposition of that numerical grid.
- Be able to easily extend the amount of data transferred between each model.
- Compute additional fields as needed by the component models.
- Interpolate data from one numerical grid to the other.

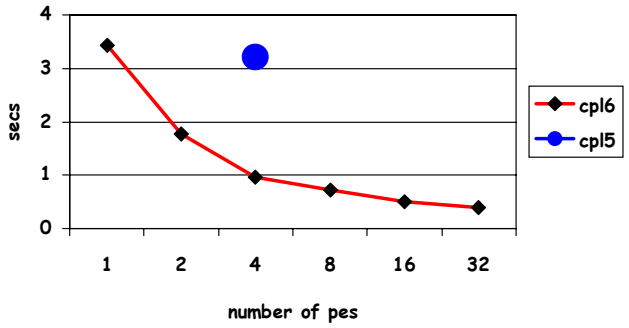


Fig. 2. Comparison of CPL5 and CPL6 coupler performance on ocn -> atm grid interpolation.

Performance of CPL6 compared to CPL5 is shown in Figure 2. It also shows the total time needed to do 120 interpolations of nine fields from the ocean grid to the atmosphere grid. When the number of processors is equal CPL6's distributed memory method, using MCT's *MatrixVectorMultiply* routine and the distributed weights and data, is almost three times faster than CPL5's threaded routine. Further speedup is available in CPL6 when the total number of processors is increased.

In support of the CPL6 effort, the MCT team has instrumented the latest in-house version of CCSM with the MPE profiling library. The output of this library showing a simulated hour of integration of CCSM on the IBM SP4 is shown in Figure 3.

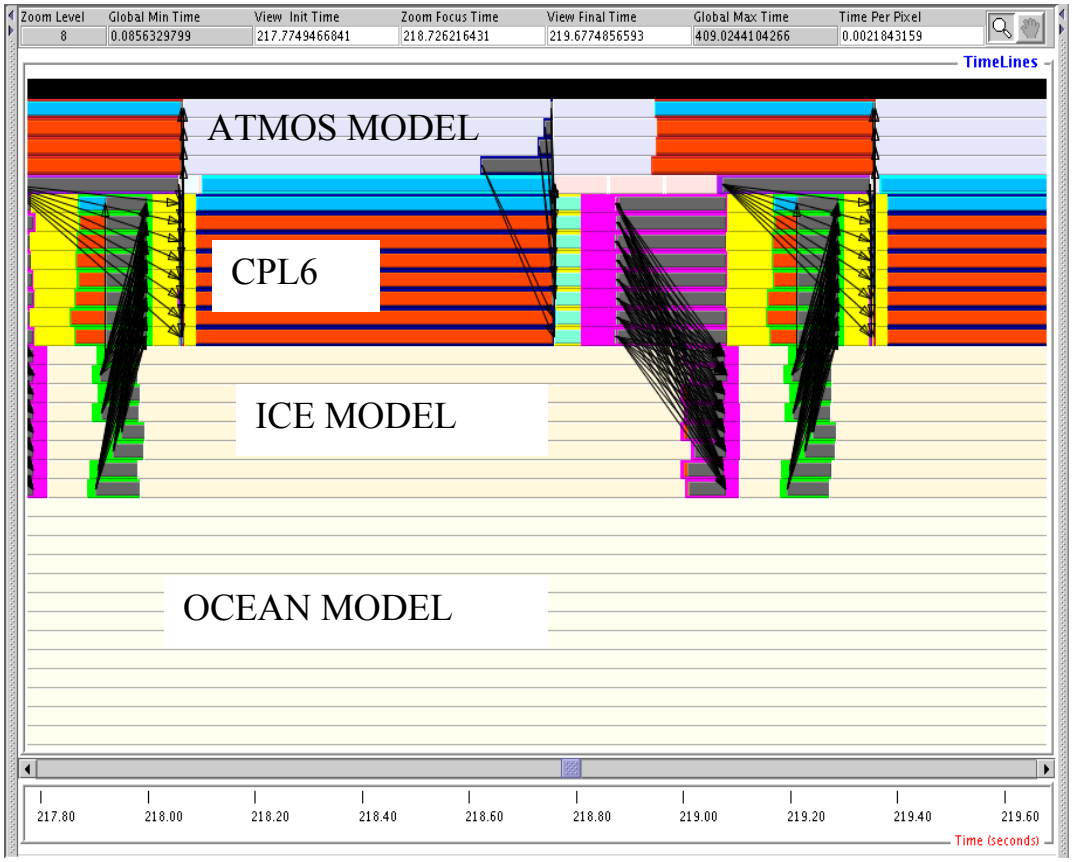


Fig. 3. Model interaction within CCSM during one simulated hour with data flow between models and the coupler indicated by arrows. Blue and Red colors indicate wait time. As can be seen, the coupler itself introduces very little compute time into the system. It is frequently waiting for the other models. Most of the waiting in the system is caused by unavoidable data dependencies created by the physical relationships between the models.

4.2 MCT (J. LARSON, R. JACOB, E. ONG)

MCT provides data storage classes, MxN communication methods, and decomposition descriptors that support any conceivable grid/decomposition pairing in a coupled system. MCT also provides grid interpolation routines via a matrix-vector multiply and includes datatypes for storing the weights calculated by the SCRIP program. Since the start of SciDAC, MCT developed several additional modules and methods to support CCSM. First, we added support for Parallel Climate Model (PCM)-style integration of CCSM. A major new module added to MCT under SciDAC includes support for spatial integration and averaging. This new module supports: (1) unmasked spatial integration and averaging, (2) masked integrals and averages using an arbitrary number of integer and real masks, and (3) paired masked and unmasked integrals to enforce conservation of fluxes. Also added to MCT was a merge facility for combining flux and state data from multiple components for use by another component. A high-level parallel matrix-vector multiplication controller was added. We have also added new non-blocking versions of MCT's MxN data transfer routines. To measure MCT's performance in CCSM, we instrumented CCSM with the MPE library (see above).

A public MCT release was made in November of 2002. We have timed the performance of MCT by itself and with cpl6 (see above) and ported MCT to several platforms including ones on which CCSM is not yet running (OSX, SunOS, Earth Simulator, Cray X1). We also provide a 250+ page API document generated automatically from self-documented code and PROTEX commands.

In addition to MCT's use in CPL6, MCT has been used to create a version of the Weather Research and Forecast (WRF) model's input/output (I/O) API for model coupling. This version of the API has been used to couple WRF to the ROMS ocean model.

Although the delay in completing CPL6/MCT was significant, the anticipated demands on dynamic load balancing and multi-dimensional fields have not appeared. We are thus well positioned to meet the new objectives within the remainder of the SciDAC project.

4.3 MPH (Y. HE AND C. DING)

MPH provides the following capabilities: component name registration, resource allocation, communicators for inter-component communication, inquiry on the multi-component environment, and standard in/out redirect. It supports four integration mechanisms:

- Single-Component Multi-Executable (SCME) — CCSM mode (MPH1)
- Multi-Component Single Executable (MCSE) — PCM mode (MPH2)
- Multi-Component Multi-Executable (MCME) — most flexible mode (MPH3)
- Multi-Instance Multi-Executable (MIME) — ensemble simulations (MPH4)

The SCME mode is used in the current CCSM. The handshaking process is done by simply invoking the MPH_components_setup () subroutine.

MPH also provides the MCSE mode, which supports the integration method used in PCM. The most general and flexible mode of MPH is the MCME mode which occurs when a model contains multiple executables and one or more executables contains multiple components, e.g., the ice and ocean models are in a single executable. A large variety of coupled systems can be constructed this way.

Multi-instance executable is another type of execution supported by MPH. The same executable is replicated multiple times (multiple instances) on different processor subsets. There is no limit of the number of instances. This enables running multiple ensembles simultaneously as a single job, and ensemble averaging be done on the fly. This eliminates large data output and storage for post-processing averaging and also enables nonlinear ensemble statistics impossible to compute at post-processing step.

In MPH, a component name registration file is needed to specify the components and their processor allocations. This registration file is only specified at runtime, providing flexibility for the software design.

Further functionality available in MPH: (a) Joining two components. A joint communicator between any two existing components could be created. (b) Inter-component communications. MPI communication between local processors and remote processors is invoked through component names and the local id. (c) Other functions are also provided for querying the global-id, local-id, number of total components, component name, local communicator and number of executables, etc.

MPH currently works on IBM SP, SGI Origin, HP Compaq clusters, Cray T3E, and PC linux clusters. It is used in the CCSM as a utility in model coupling toolkit. It is also being adopted in Colorado State University's icosahedra grid coupled climate model. Other users are NCAR's WRF (successor of MM5), a group in Germany developing a coupled climate model, and a group in UK/ECMWF for ensemble simulations. For more information see http://www.nersc.gov/research/SCG/acpi/MPH.

4.4 INCORPORATING MLP INTO CCSM (JIM TAFT)

The Japanese ES system, and its reported results on various earth science computations, stimulated interest in examining the performance of the CCSM2.0 climate model on shared memory systems using the MLP parallelization strategy. The CCSM2.0 model uses CPL5 which contains only OpenMP parallelism and allows only a single MPI process. The coupler in CCSM transfers a significant amount of data, and CPL5 can be a severe bottleneck in the speed of the overall computation. Much of the bottleneck can be attributed to inefficiencies created in communicating between the single-MPI-task coupler and the multi-MPI-task physical models.

Table 2. Performance of MPI and MLP versions of CCSM2.0 on SGI Origin 3000 and IBM Power3 platforms. Compute time (days) for a 1000-year simulation

Computer	Location	GF/PE	#PEs	Comm	Days	Speedup	Years/Day
SGI O3000	NASA Ames	1.2	208	MPI	318	1.0	3.1
IBM Power3	NERSC	1.5	128	MPI	200	1.6	5.0
SGI O3000	NASA Ames	1.2	208	MLP	73	4.4	13.7

The current effort to increase the performance of CPL5 involved rewriting the coupler communication routines and the corresponding communication routines in each of the sub-models to use MLP. This effort was completed, and the results of the effort are shown in Table 2. Three runs were done using the standard T42 atmosphere and 1 degree ocean inputs distributed with the model. The first line in the table shows the results for the original code as executed on the Origin. The second is the reported result for the original code on the Power3 cluster performed at National Energy Research Scientific Computing Center (NERSC). The bottom line is the result for CCSM2/CPL5 after its conversion to MLP-based inter-module communication. Origin executions utilized 208 CPUs. The IBM execution utilized 128 CPUs.

As can be seen the original Origin result is substantially slower than the IBM Power3 result. This is due to the relative speed of the CPUs and the relatively lower Origin memory bandwidth. With the addition of MLP coding the communication bottleneck ceases to exist, and the resulting code executes 4.4 times faster on the same machine (SGI) and almost three times faster than previously achieved on US hardware.

Work continues on this code. Expectations are that at least another factor of two exists for the same CPU count. In addition, scaling to larger CPU counts will be possible, once the remaining MPI-based intra-module communication is exchanged for MLP equivalents.

Plans

As above, Plans can be separated into those for CPL6, MCT and MPH.

CPL6

Future development of CPL6 includes work to configure CCSM as a single-executable and hardening and optimizing the CPL6 classes. For the remainder of the SciDAC climate project, the priorities are:

- Prepare CPL6 for public release with the new CCSM.
- Develop a vector friendly version of CPL6 and MCT to support a vector friendly version of CCSM.
- Develop a single executable CCSM with concurrent integration.
- Improve the ice-coupler communication speed.
- Find optimum configurations for various resolutions of CCSM on supported platforms.
- Increase the performance of CPL6/MCT on all supported platforms.
- Provide more methods for CPL6 classes.
- Fold selected CPL6 datatypes/methods into MCT.
- Continue porting CPL6/MCT to new platforms.
- Develop a sequential-execution version of CCSM.

We will also modify CPL6 to optionally support the use of a subgrid orography scheme within the atmosphere and land models. This will require exchanging a large number of additional fields between the coupler and the land and atmosphere models. The MCT development team will be assisting NCAR scientists in all of these tasks.

MCT

The CPL6 goals will be met largely through the development of a second-generation version of MCT. This new version (2.0) will embody numerous features that will improve the overall performance and scalability of MCT-based applications, and will increase dramatically the flexibility of the CCSM modeling system.

The main development activities to improve MCT's performance are: adding support for OpenMP parallelism in the more numerically-intensive parts of the toolkit; platform-specific tuning; optimization of the collective communications operations in MCT, especially intermodel exchange of data decomposition information.

The incorporation of OpenMP support in MCT had been delayed in favor of more urgent development activities. Now that a distributed-memory version of CPL6 based on MCT has been stabilized and tested, we are ready to add OpenMP parallelism to the more compute-intensive portions of MCT—the matrix-vector multiply (interpolation), time-averaging and accumulation, global integrals and averages, and the merge facility. These additions may improve scalability of MCT applications to large numbers of processors, and will also be useful in addressing the multiple levels of parallelism present on the Cray X-1.

Platform-dependent tuning of MCT may revolve more around the tuning of a vector-friendly version of the toolkit. The original target platform for MCT was commodity-based multiprocessors where cache re-use is the primary single-processor performance issue. MCT currently performs well on these platforms, as we have demonstrated earlier in this report. We will work to incorporate vector modifications to MCT provided by scientists at NEC/Fujistu into the source code without sacrificing the micro-processor performance.

Collective communications in MCT fall into three categories: MPI collective communications calls used in global sums; parallel data transfers and redistributions; intercomponent exchange of domain decomposition information.

We assume that most MPI implementations use an optimal scheme such as the "butterfly" algorithm for the MPI_ALLREDUCE() we use in our global integration routines, but intend to test this assumption.

The parallel data transfer schemes in MCT are designed to send only the data that is required for the target component's purposes; that is, the messages are packed. The intracomponent parallel data redistribution scheme in MCT also uses a message-packing, coupled with logic to prevent self-messaging. These schemes perform well, but are sensitive to differences in the source and target decompositions. We intend to study this problem carefully to gain a better understanding of these sensitivities, and explore possible solutions that may boost overall data transmission performance. The final collective communications challenge in MCT is the exchange of domain decomposition data between components—a process we call *handshaking*. Currently, this operation is implemented in MCT as a point-to-point transfer between the components' respective root processes, followed by a push of data out to the rest of the components' processor pools. This process is effectively serial, and has been treated as a start-up cost, a strategy that is viable in the current CCSM. We will need to explore parallel handshaking to support dynamical load balance in the component models. Note here that we are referring to a dynamical load balance scheme in which each component model has a fixed number of MPI processes (we will discuss the more general case shortly). Dynamical load balancing in CCSM's component models will require periodic updates of domain decomposition information to the coupler from the models that employ this technique. This update process could potentially impose a performance bottleneck. The solution to this problem is the parallelization of the handshaking facilities in MCT. Another handshaking challenge is posed by dynamically varying physical grids. This challenge will be met through the parallelization of the transfer of physical grid description information (as embodied in the MCT GeneralGrid datatype).

<u>MPH</u>

A key requirement for CCSM is to have a single-executable version such that smaller research groups can run CCSM effectively on Linux clusters with limited resources. The single source code policy demands that the single-executable version coexist with the multi-executable version. This involves some fundamental restructuring of top level subroutines (subroutinization of component program models), and appropriate restructuring of MPH.

Another highly desirable feature for CCSM is to be able to switch between data models, active component models, and dead models (mock model, for place-holder purpose) at run-time. This work is better carried out simultaneously with the above model development. Together, these represent a major reengineering of the CCSM code structure so that it supports multiple model integration mechanisms that MPH provides.

Several additional desirable features of model integration mechanisms of MPH are: (a) flexible way to handle shared memory processor (SMP) nodes, i.e., recognizing a 16-cpu SMP node could be carved into different number of MPI tasks; (b) dynamic component model processor allocation or migration; and (c) an extension of MPH to do model integration over the grid. (d) C/C++ version of MPH.

5.0 COMMUNITY ATMOSPHERE MODEL (TOPIC COORDINATOR: J. DRAKE)

The atmospheric model is arguably the most complex of the component models. The effort on performance and extensibility is largely covered in the software engineering chapter. In this section we discuss the efforts to increase parallelism and scalability of the dynamical cores. The FV dynamical core developed by S.J. Lin and Ricky Rood is the subject of the first section with more standard Eulerian –Spectral and Semi-Lagrangian – Spectral dycores. The effort to increase parallelism is a necessary preparation for higher resolution studies that require significantly higher levels of computation. Two sections in this atmospheric Chapter deal with high resolution studies. The first uses a novel technique to increase orographic detail and realism in the simulation. The second addresses the scientific performance of the CAM2 as resolution is increased. A final section deals with the issue of I/O in the CAM. A parallel I/O library has been developed which interfaces with the parallel NetCDF library being developed by the SciDAC Scientific Data Management ISIC.

5.1 LIN-ROOD OPTIMIZATION (A. MIRIN)

Background and Objectives

Our focus has been on performance optimization of the NASA/Goddard Space Flight Center (GSFC) FV dynamical core, with the order of attack being:

(1) MPI performance of the 1-D (latitudinal) decomposition.
(2) MPI performance of the multi-2D decomposition methodology.
(3) OpenMP performance.

Additionally, there has been an ongoing effort in software engineering. This work has been carried out in collaboration with Will Sawyer of NASA/GSFC.

Progress

Completion of area (1) was reported last December. Work in area (2) is virtually complete, the one exception being full implementation of MPI2 one-sided communications (see below). This effort has focused on the transposes connecting the primary latitude/vertical decomposition used for most of the FV dycore and the secondary longitude/latitude decomposition used for the remapping of the Lagrangian surfaces (this secondary decomposition is used to connect to the physics module as well). These transposes make use of the *Mod_comm* (NASA/GSFC) library for their execution and the *Pilgrim* (NASA/GSFC) library for communication pattern setup. One has the choice of transposing data in temporary, contiguous buffers, or bypassing the temporary buffers in favor of MPI derived types. The latter approach appears to be very slightly favorable on the IBM. One also has the choice of using traditional MPI two-sided communications (e.g., *send*, *receive*) or MPI2 one-sided communications (e.g., *put*). One-sided communications can be quite favorable on the SGI but are decidedly inferior on the IBM. The one task yet to be completed is support for MPI2 one-sided communications in conjunction with MPI derived types; this would be most relevant to the SGI. Much of this coding was in place and working last December; recent efforts have focused on modularization and optimization.

There are several motivations for examining OpenMP performance. One is that with the 2D decomposition, one of the decomposition directions (vertical) is the same as the primary OpenMP direction, thereby reducing the degree of attainable parallelism. Also, with the domain decomposition methodology at moderate resolution limiting the number of parallel tasks,

improvements in OpenMP parallel efficiency would allow use of a greater number of threads (hence processors), thereby reducing the time to solution. For tracer advection we parallelize with respect to both tracer index and height. Increased thread usage for other parts of the dycore is not as straight-forward; hence, we are examining parallelization over multiple directions simultaneously through use of nested constructs. Nested OpenMP is presently supported on HP/Compaq and IBM platforms, although IBM's present implementation is non-standard and not well publicized.

Software engineering efforts are focused on source code streamlining and support for vector platforms. A number of improvements have been made, including removal of outdated options and code variables, and modularization in general. An important task yet to be accomplished is merging of distinct coding paths relating to the 1D and 2D decompositions. Vectorization improvements are being carried out largely by Dave Parks of NEC.

Performance evaluation of CAM2 with the FV dycore has been carried out on the IBM Seaborg machine at NERSC. Seaborg has 16 power-3 processors per (Nighthawk 2) node. Cases have been carried out using version cam2_0_2_dev37 with mpi derived types for the transpose communications. Figure 4, FV-1, shows performance of CAM2 on the 2x2.5x26-level mesh as a function of processor count. The red curve relates to the 1D decomposition and the green curve to the multi-2D decomposition. All cases utilize fourMPI tasks per node and 4 OpenMP threads (with static scheduling) per task, with the physics decomposition corresponding to the dynamics longitude/latitude decomposition. Figure 5, FV-2, shows CAM2 performance on the 1x1.25x26-level mesh, and Figure 6, FV-3, the 0.5x0.625x26-level mesh (but with 8 OpenMP threads per task instead of 4, due to memory requirements). In all cases the multi-2D decomposition technique (as compared to the 1D decomposition) offers the possibility of doubling the throughput. For WACCM configurations (which have a much greater number of vertical levels and constituents), the increased throughput offered by the multi-2D decomposition method is substantially greater, as reported in the December 2002 progress report.

We now examine scalability of the CAM2 components. Figure 7, FV-4, and Figure 8, FV-5, show a breakdown of major code sections for the 1x1.25x26-level resolution for 16 (4x1 decomposition) and 1344 (48x7 decomposition) processors, respectively. The FV dycore itself scales reasonably well. For the 1344-processor case, the fractional time taken up by the main dynamics is virtually the same as that for the 16-processor case; the fractional time of the geopotential calculation (which strongly couples the vertical levels), although doubling, is held to 14%; the tracer advection and Lagrangian remapping total 6%; the dynamics to physics coupling is only 6%, and the fractional transpose time is a mere 4%. The main bottleneck appears to be the land surface model, which takes up 39% of the total computer time, as opposed to 7% for the 16-processor case.

We also examine performance with respect to the OpenMP scheduling mechanism. Default static scheduling incurs the least OpenMP overhead, whereas dynamic and guided scheduling offers the possibility of better load balance. Carrying out a full range of cases for the 1x1.25x26-level resolution, we find that guided and static scheduling give almost identical performance. An isolated case of dynamic scheduling shows nearly identical performance as well.

Plans

Areas of ongoing/future work include: (1) investigation and evaluation of nested OpenMP constructs in the main dynamics module, (2) creation of a unified coding path for 1D and 2D decompositions, (3) implementation of vector constructs, and (4) implementation of MPI2 one-

sided communications with MPI derived types in the transposes. We are collaborating with the NASA ESMF effort in the creation of an FV module (Will Sawyer, contact).

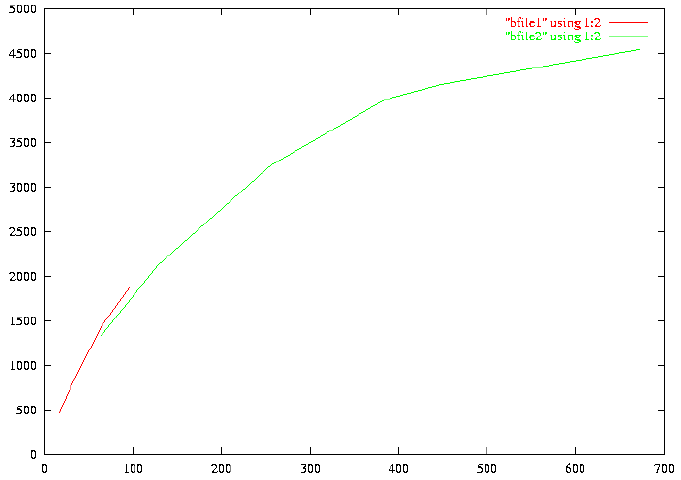


Fig. 4. FV-1 Performance of CAM2 with Finite volume dycore at 2x2.5x26-level resolution on NERSC IBM Seaborg, in terms of simulated days per computer day versus processor count.

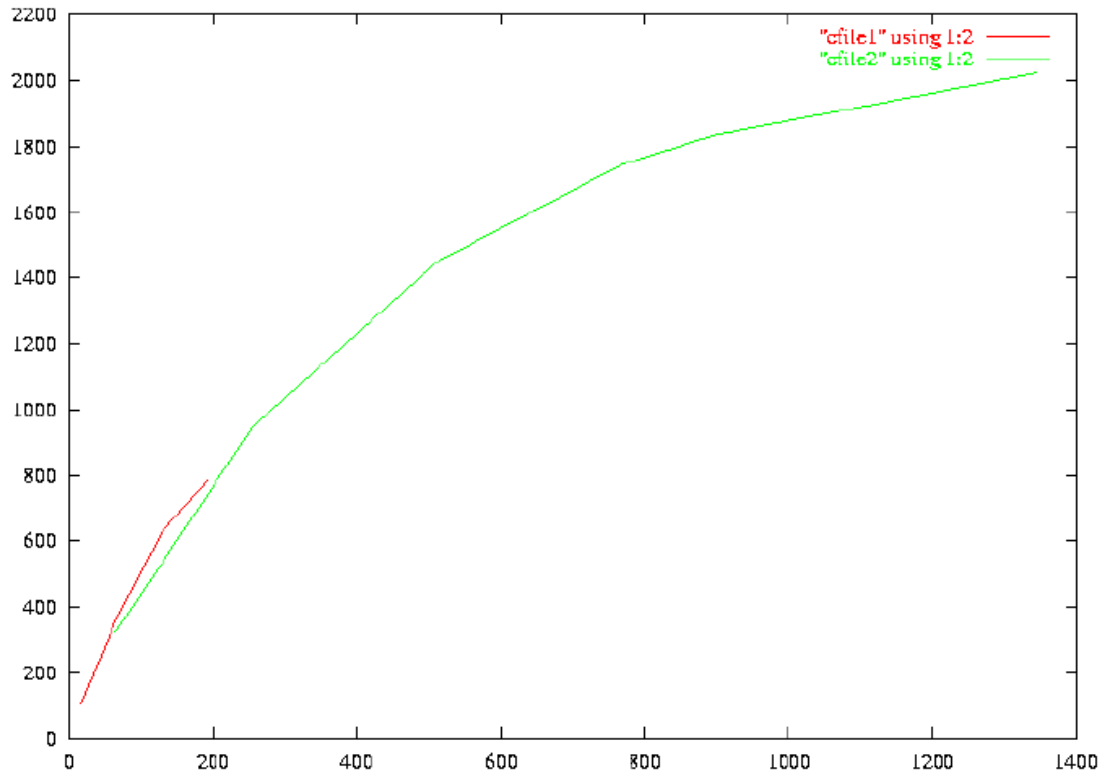


Fig. 5. FV-2: Performance of CAM2 with finite volume dycore at 1x1.25x26-level resolution on NERSC IBM Seaborg, in terms of simulated days per computer day versus processor count.

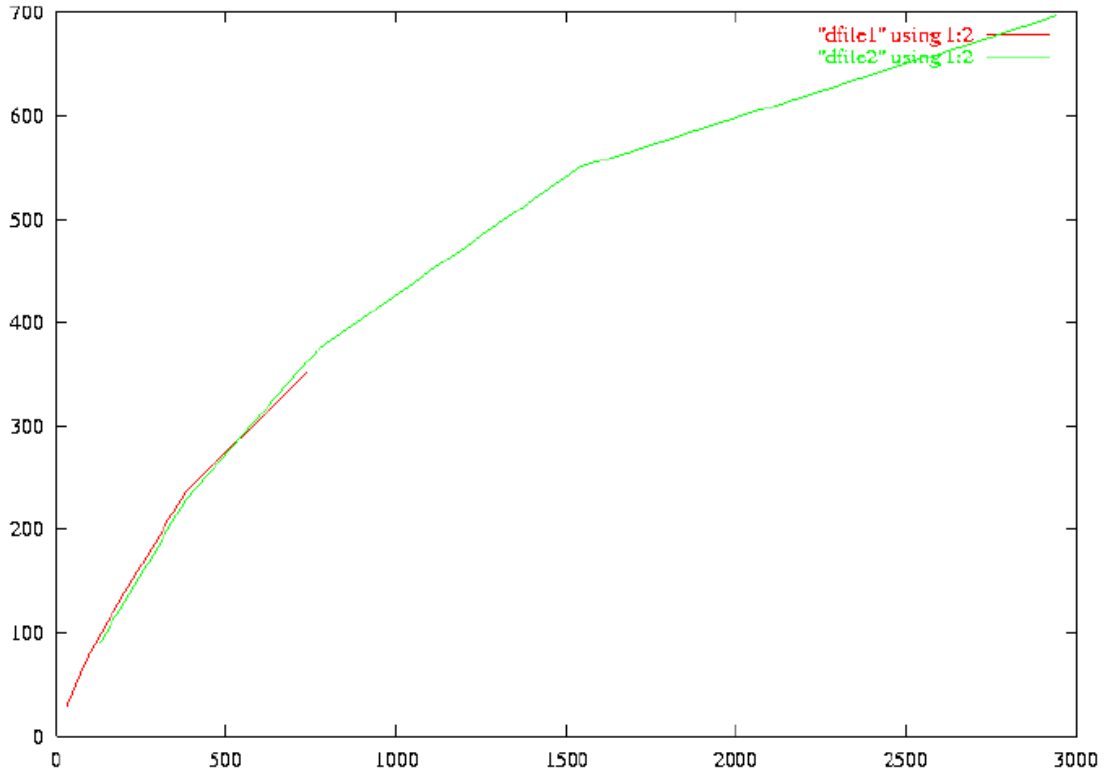


Fig. 6. FV-3: Performance of CAM2 with finite volume dycore at 0.5x0.625x26-level resolution on NERSC IBM Seaborg, in terms of simulated days per computer day versus processor count.

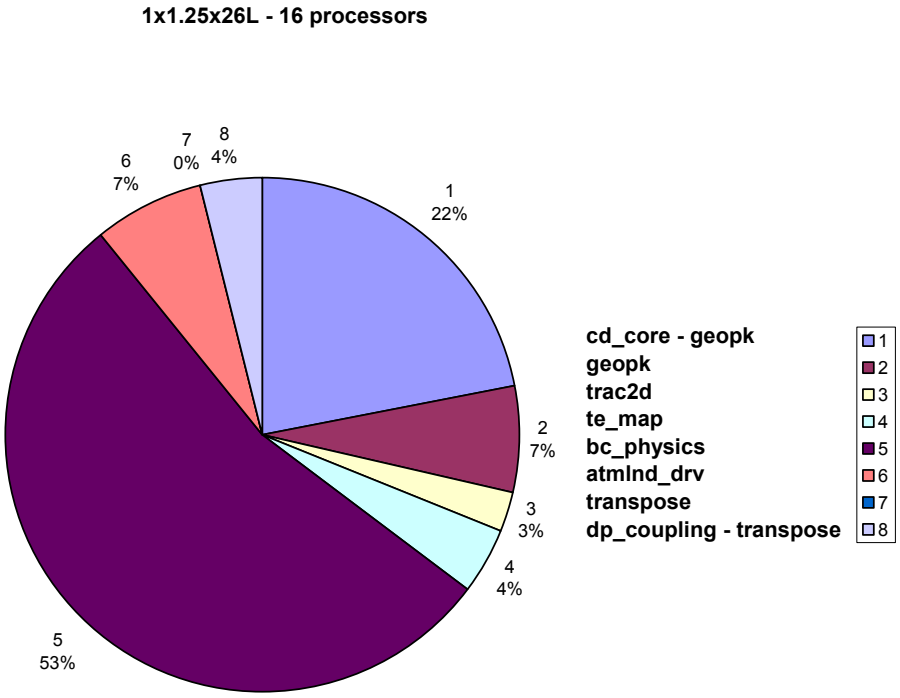


Fig. 7. FV-4: Distribution of work for 1x1.25x26-level case using 4x1 domain decomposition.

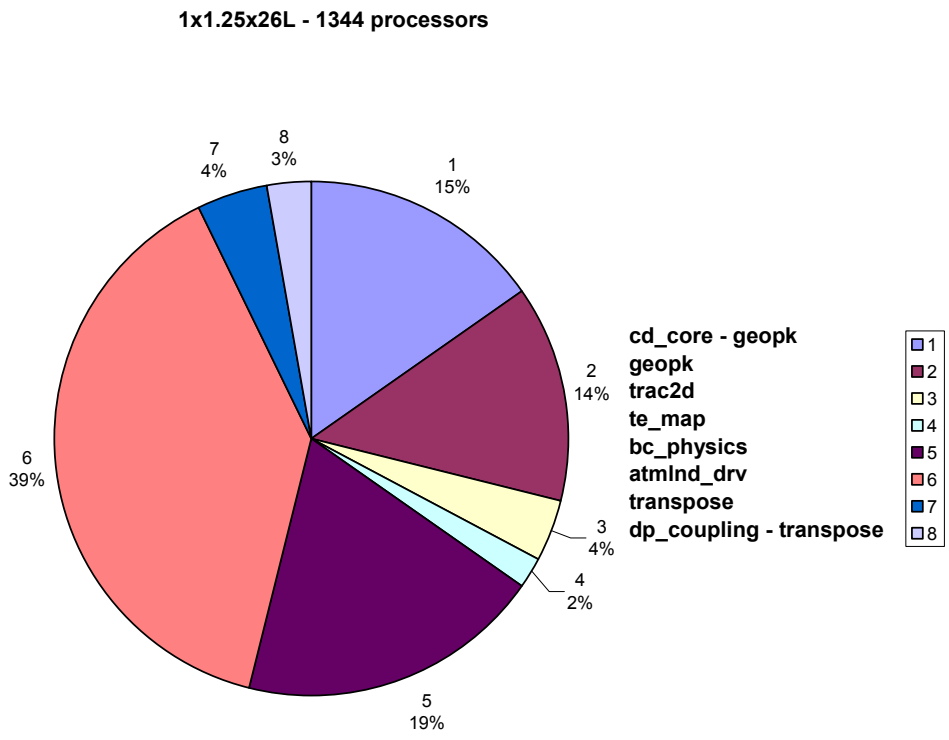


Fig. 8. FV-5: Distribution of work for 1x1.25x26-level case using 48x7 domain decomposition.

5.2 SPECTRAL ATMOSPHERIC DYNAMICS, PHYSICS CHUNKING AND LOAD BALANCING (P. WORLEY AND J. DRAKE)

Background

The Eulerian - Spectral dynamical core is still the workhorse dynamics for climate change simulation. The SciDAC project began with the spectral dynamics only partially separated from the "physics" parameterizations. The ACPI AG project provided a preliminary design that we have continued to work with during the SciDAC project. At this point the software engineering and design has resulted in a separation of the data structures that allow work on other aspects of the parallel algorithms.

Objectives

The objectives for the CAM spectral dycores focus on the development of high resolution models that perform well on scalable supercomputers.

Table 3. CAM objectives and tasks

	Status	Deliverable	Date
Performance optimization			
Cache-friendly physics	Done	CAM2	3Q02
Cache-friendly dynamics FV	Done	CAM2	3Q02
Cache-friendly dynamics EUL&SLD	Done	CAM2	2Q03
Load balancing	Done	CAM2	3Q03
Adaptation to new hardware	ongoing	CAM2	
OpenMP FV optimization	In progress	CAM2	4Q03
Transposes in FV	Done	CAM2	2Q03
ModCOMM : MLP and MPI2	In progress	CAM2	1Q04
dynamic load balancing scheme	Not needed		
Parallel Decompositions			
2D decomposition	Done	CAM2	4Q02
Physics Chunking	Done	CAM2	4Q02
Eulerian and SLD Spectral	Ongoing		
Finite Volume	Done	CAM2	3Q02
Block dycore Interface	Ongoing	Prototype	3Q03
Vectorization			
Dynamics and Physics Vectorization	In progress	CAM2	4Q03

This table shows some changes in the original objectives, partly because of the advent of vector computers and the shifting landscape with computational frameworks. The block dynamics interface has been implemented but continues to change as ESMF data types and structures are employed.

Progress

With the separation of the physics and dynamics and the incorporation of physics chunking the issue of load balance and optimal configuration are raised. Performance measurements showed that the optimal chunk size was different on each machine and that performance gains on cache based machines were possible by setting the chunk size to less than 64 (the default setting for a T42 latitude slice). For the IBM SP3, the optimal size is 16-32 while the Compaq is 8-16. The implementation of load balancing of the physics calculations (especially the short wave) was taken as a high priority due to the potential benefit to performance. With load balancing, each physics chunk is assigned a nearly equal amount of work. The extra granularity available from chunking the physics allowed an increase in the number of processors used in the standard atmospheric decomposition. With multiple chunks per processor, the shared memory parallelism on a node is increased and we are able to easily take advantage of the hybrid distributed-shared memory-programming model that has been adopted for the CAM. The configuration now

recommended is to distribute the dynamics across MPI processes and the physics across shared memory processors as wll as MPI processes. So the Eulerian or Semi-Lagrangian dynamics running a 1D decomposition on 32 nodes (at T42) can take advantage of a factor of 4 – 8 more processors. Thus the scalability of the CAM has been improved.

Work on the parallel algorithm of the spectral dynamical cores has also resulted in performance improvements. Specifically, the spectral algorithm no longer duplicates spectral coefficients across all the processors; spectral calculations are now m-decomposed. With this change, communication (in the REALLOC routines) has been reduced and better scalability is obtained. These, together with elimination of the gather-scatter logic in the land models interface with the physics, have resulted in a factor of 5 increases in the achievable throughput of the CAM.

In the last year, we have taken a course to include vector-friendly configurations in the CAM. The target platforms for this work are the Cray X1 and the NEC SX-6. Coordination of this new effort started with a workshop held in Boulder (February 2003). The SciDAC effort began looking at restructuring code to exploit vector performance and has now evolved to developing a full implementation of the CCSM2 on these platforms. This is the most rigorous test to date of our software practices and the ability to achieve performance portability. This work is progressing under the assumption that it is unacceptable to degrade performance on the cache based machines that provide the preponderance of cycles to climate simulations.

The new data structure includes a compile time specification of the maximum size of the basic data structure (for serial performance efficiency) as well as a runtime determination of the actual size, to support the run-time specification of processor counts, MPI tasks, and OpenMP threads per MPI task. The size parameter can also be used to improve vectorization or cache locality, and to expose additional MPI or OpenMP parallelism.

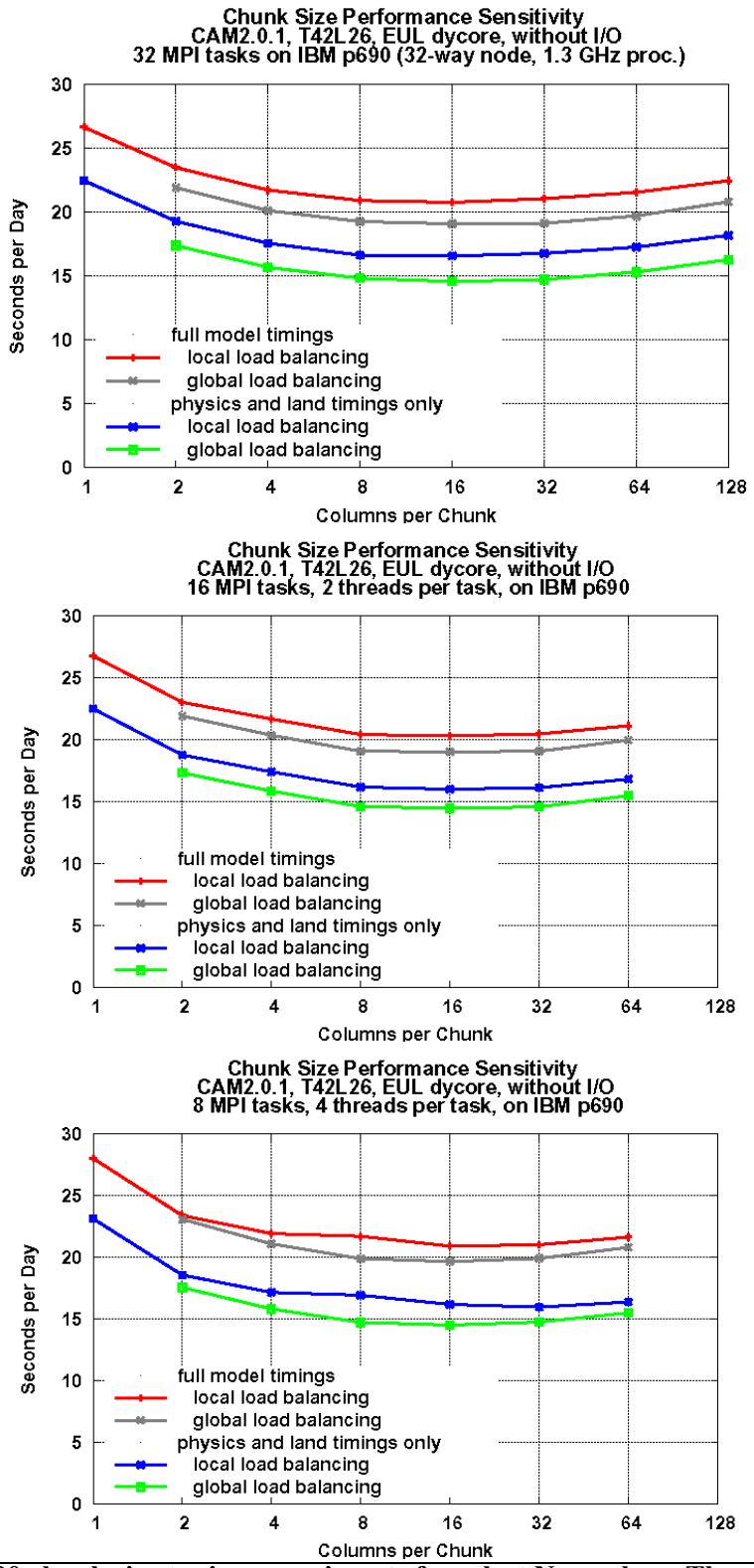


Fig. 9. IBM p690 chunk size tuning experiments from last November. These data should be qualitatively correct. These are on a single 32-way node, so the load-balanced option looks better than it would if sending messages out into the Colony switch.

We continue to maintain the physics chunking, helping NCAR implement the data structure in new physics routines (e.g., sulfur cycle, ozone, and other aerosols).

To address the load balance issues in the parallel code we designed and implemented a number of different static load balancing schemes for the physics, three of which are part of the official release. The schemes vary from load balancing within the same domain decomposition used by the dynamics (requiring no interprocessor communication) to an almost exact load balance that requires additional interprocessor communication.

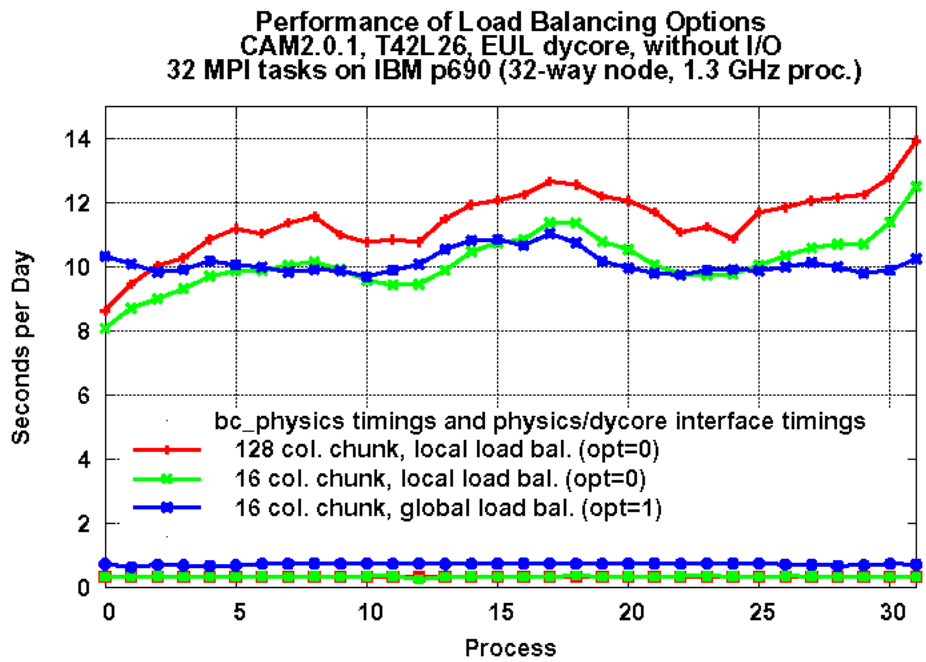


Fig. 10. This comparison is between opt=0 and opt=1. The data for the opt=2 is not available at the moment. These data are from February, but should be indicative of the current performance tradeoffs. opt=1 again looks better than it would if it required internode communication.

The current 1D parallel decomposition was redesigned for the Fourier and spectral spaces of the spectral dynamical cores, blocking the transforms were implemented for improved serial performance and to decrease interprocessor communication. This also had the effect of improving load balance, and decreasing memory requirements.

In preparation for the move to a T85 spectral resolution, we have investigated the memory requirements of CAM. The initialization routine inidat has been completely restructured, significantly reducing the memory requirements. Additional memory was also eliminated from the Eulerian spectral dynamical core. This allows an improved footprint for the higher resolution model.

In collaboration with PERC and with the CCSM Software Engineering Group, we have been porting and determining optimal tuning parameter settings for CAM on a number of different platforms. We have also been tracking the performance evolution of CAM and identifying remaining performance problems.

5.3 PERFORMANCE STUDY OF HIGH-RESOLUTION CAM (W. YANG, M. WEHNER AND C. DING)

We have examined high-resolution performance of CAM2.0.1 using the T170L26 resolution as well as CCM3.10 using the T170L18, T239L18 and T340L18 resolutions. We note that memory allocation in both CAM2.0 and CCM3.10 is inappropriate for high-resolution simulations since a large chunk of memory required does not scale with MPI process count (1.80 GB out of total 7.37 GB per process in case of the T170L26 Eulerian CAM2.0.1; see the Figure 11). On the IBM SP at NERSC, we have tried various partitions of the SP nodes among MPI tasks and OpenMP threads, and noted that the best performance with CAM2.0.1 is achieved when eight MPI tasks are specified for each SP node. Compared to T170L18 CCM3.10 runs, T170L26 CAM2.0.1 runs require more run time because there are more vertical levels involved and more physics calculations performed, but need less message passing time and, therefore, exhibit a better parallel scaling.

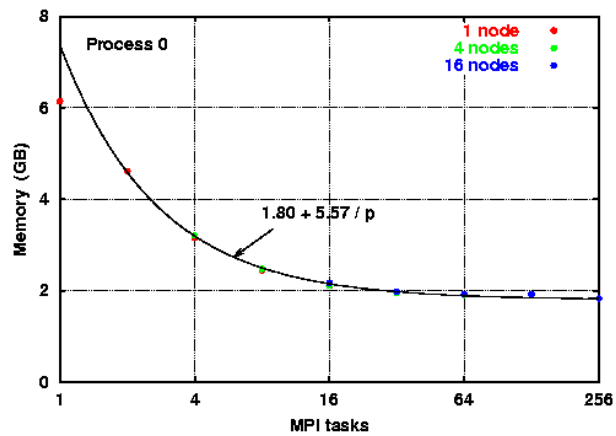


Fig. 11. Memory usage of CAM2.0.1 at T170L26 resolution on per processor basis.

Plans

Ongoing work includes a number of interprocessor communication optimizations, both within the dynamical core and between the physics and the dynamical core. Some of these are system-specific, and the intent is making these runtime options. We also intend to include or reimplement support for SHMEM and Co-Array Fortran, so as to further reduce communication costs on systems that support these alternatives to MPI.

Other ongoing work includes the implementation of a two-dimensional decomposition in the spectral dynamics.

Vectorization of CAM has begun in collaboration with Cray, NEC, CRIEPI, Fujitsu, and NCAR. This work has a target completion date of January 1, 2004.

PERC Performance Evaluation Research Center Collaboration

Pat Worley, Oak Ridge National Labortory
(http://www.csm.ornl.gov/evaluation/CAM)

The following graphs describe the performance evolution of the Community Atmospheric Model (CAM) when run with the spectral Eulerian dynamical core. Over the past few years, a number of modifications have been made to the CAM with the goal of improving performance. With one exception, described below, these changes were introduced as options that can be disabled. This allows us to examine the impact of these changes and the evolution of CAM performance. The data presented here was collected on the IBM p690 cluster at Oak Ridge National Laboratory and on the HP/Compaq AlphaServer SC cluster (lemieux.psc.edu) at Pittsburgh Supercomputer Center during September and October of 2002. Version 2.0 of CAM was used for all but one of the experiments. The problem size was T42L26, corresponding to a 128 by 64 by 26 computational grid.

CAM has two primary computational phases, the dynamics and the physics. The physics has the higher serial complexity for the target problem resolutions, but it is also easily parallelized. Parallel algorithms for the dynamics require significantly more interprocessor communication than for the physics. In addition, the parallel implementation of the spectral Eulerian dynamics currently supports only a 1-D decomposition of the computational domain, limiting the number of usable processors to 64 for T42L26. By using hybrid MPI/OpenMP parallelism, more processors can be applied to the physics than to the dynamics, improving scalability.

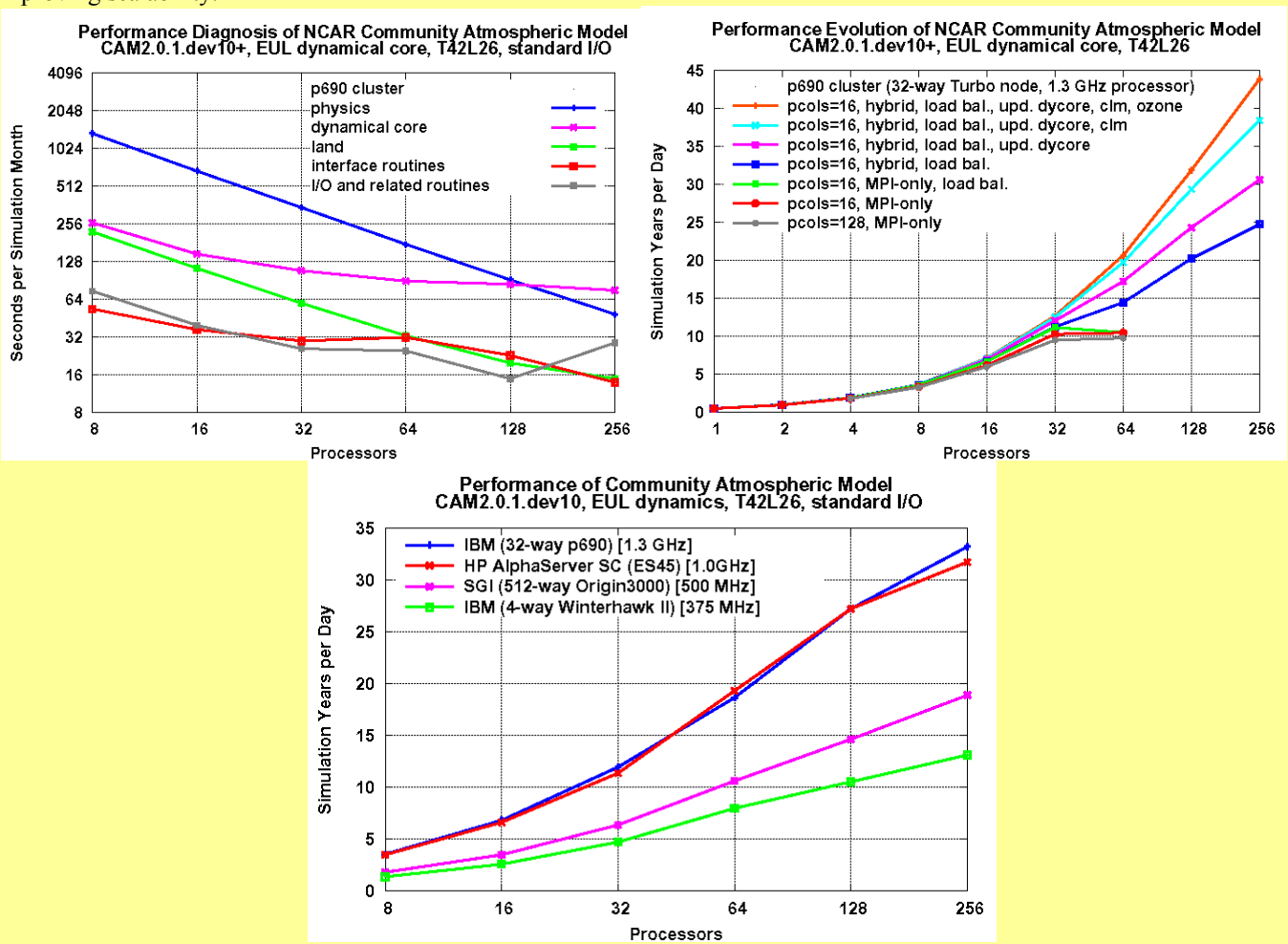


Fig. 12. a) The performance evolution of CAM (Eulerian dynamical core on Cheetah (IBM p690), up to 2_0_1.dev10 + ozone update, no i/o), b) performance diagnosis (Eulerian dynamical core, up to 2_0_1.dev10 + ozone update, with standard i/o), c) performance benchmarking of the Eulerian dynamical core on Cheetah, Lemieux, Guyot, Eagle.

5.4 SUBGRID OROGRAPHY SCHEME (S. GHAN AND T. SHIPPERT)

Background

In regions with complex terrain, topography can produce an orographic signature to climate that is far too detailed to be explicitly resolved by global climate models for at least the next ten years. A computationally efficient alternative to using extremely high grid resolution is to use the Leung and Ghan (1995, 1998) subgrid orography scheme. In that scheme, surface elevation is classified into a modest set of elevation classes, a high-resolution surface elevation dataset is used to determine the fractional area and mean elevation of each elevation class, a simple airflow model is used to determine the displacement of air parcels passing through each grid cell, conservation of energy and moisture is used to predict the temperature and moisture profiles for each elevation class, and the model column physics is then applied to each elevation class. The high-resolution surface elevation dataset can be used in post processing to distribute any model history field according to the high-resolution distribution of surface elevation. Ghan et al. (2002) have applied this scheme to a prototype of CAM2, and have evaluated its performance. Figure 13 demonstrates the performance of the scheme in the prototype of CAM2.

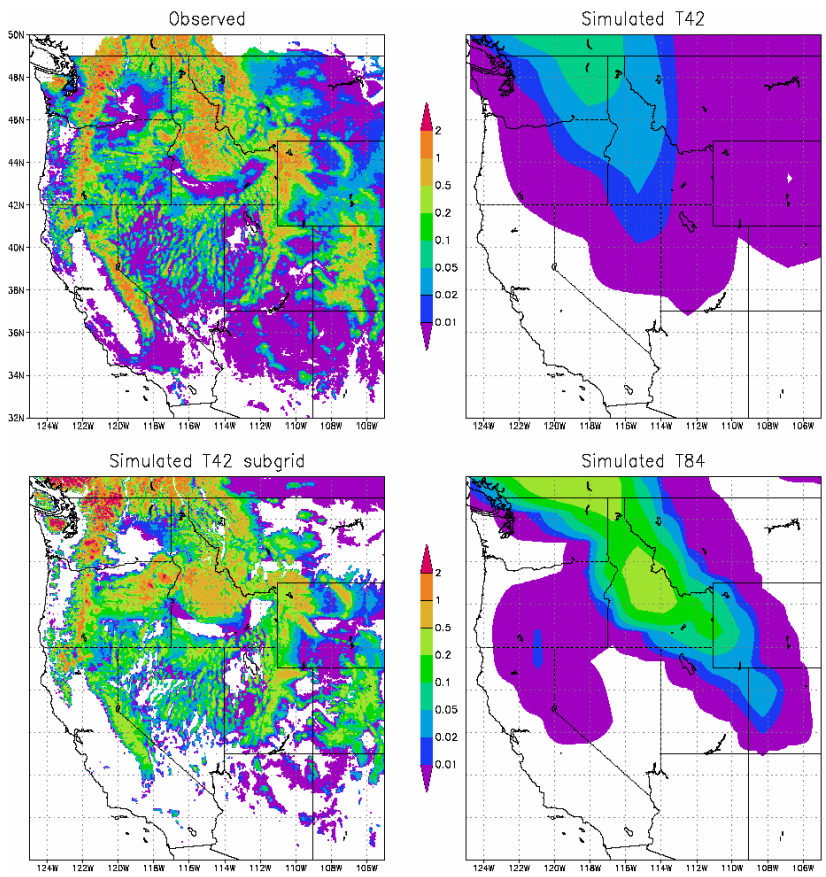


Fig. 13. Comparison of observed and simulated snow cover using the Subgrid Orography Scheme in a prototype of

In this SCIDAC project the scheme is being applied to a developmental branch of CAM2.

Objectives

The primary goal of the subgrid orography application effort is to get the scheme endorsed as permanent feature in CAM. This feature will always be treated as an option because the computational cost of the scheme (about a factor of three for the atmosphere and land models) is higher than many CAM users will be willing to pay. A secondary (new) goal is to use the scheme in IPCC scenario simulations (with ocean conditions prescribed from coupled simulations without the scheme) to produce high-resolution simulations of global climate change.

The primary goal has been expanded to support CCSM as well as CAM, because the CCSM plans are to phase out support for CAM as a stand-alone model. Moreover, application of the scheme

to CCSM will provide more refined simulations of river transport and discharge to the oceans, an important feedback on century timescales.

Warren Washington has expressed his support for this effort.

Task	Status	Deliverable	Date
Sub-grid scale orography scheme	in progress	Working code	1Q04
IPCC downscaling simulations	on hold	Publication	3Q04

Progress

To date, the subgrid scheme has been applied to both the euler and FV dynamical cores in CAM, updated to cam2_0_2_dev36. This version has the same physics as the version being used for IPCC experiments. The chunking and clumping of the atmosphere and land physics have been used to distribute the computational burden of the column physics uniformly to all processors on each node. For the euler dycore the column physics has also been distributed uniformly across nodes. This provides nearly perfect balancing of a computational load that varies ten-fold between grid cells. The cost of the dynamic-physics transpose for this load-balancing is minimized by assigning chunks to the node with the most dynamics grid cells from that chunk. For a T42 configuration with a maximum of 11 elevation classes run on eight nodes, 40% of the atmospheric physics columns are assigned to the same node as the dynamics. Short simulations have shown bit-for-bit agreement for restarts, for domain decomposition, and for load-balancing. Longer simulations demonstrate conservation of energy and moisture.

Plans

For the remainder of the project, the priorities are to:

1. Optimize the performance with the subgrid scheme by performing sensitivity experiments with different elevation classifications and different orographic relaxation timescales.
2. Document and evaluate the performance of CAM with the subgrid scheme for both the FV and euler dycores using IPCC physics. 10-year simulations will be performed at T42, T85, 2x2.5, and 1x1.25 resolution.
3. Present the performance evaluation to the land and atmosphere model working groups for their endorsement.
4. Apply load balancing to the finite-volume implementation of the scheme, for both 1D and 2D domain decomposition.
5. Perform IPCC simulations with CAM using the subgrid scheme and ocean surface conditions from CCSM simulations without the subgrid scheme.
6. Adapt the scheme to CCSM.

5.5 HIGH-RESOLUTION SIMULATIONS WITH CAM2 (P. DUFFY, J COQUARD, B. GOVINDASAMY)

Background and Objectives

We performed simulations of the global climate using CAM, developed at NCAR. We performed and analyzed simulations at spatial resolutions of 2.0 x 2.5 deg., 1.0 x 1.25 deg, 0.5 x 0.625 deg,

and 0.4 x 0.5 degrees. We first present an overall evaluation of the multi-year time-averaged simulated climate at these four spatial resolutions. Next, we examine the simulated response to El Nino at the 2.0 x 2.5 deg and 0.5 x 0.625 deg resolutions.

The simulations we analyzed were performed with version 2.0 the CAM2 [1], the fifth generation of the NCAR atmospheric general circulation model. A detailed description of the model can be found at http://www.ccsm.ucar.edu/models/atm-cam/.

The default configuration of CAM2 uses Eulerian spectral dynamics at T42 truncation, corresponding to a horizontal grid size of 2.8 x 2.8 deg. In the simulations discussed here, the model was configured using the FV dynamical core (http://gmao.gsfc.nasa.gov/sci_research/NASanCAR/). This dynamical formulation was developed at the NASA/GSFC, and is a new option is this version of the model. We performed simulations at the following four spatial resolutions: B32 (2.0 deg. x 2.5 deg.), C32 (1.0 deg. x 1.25 deg.), D32 (0.5 deg. x 0.625 deg.), and 0.4 deg. x 0.5 deg. All simulations used 32 levels in the vertical. In all of our simulations, the model was forced with prescribed sea surface temperatures (SSTs) and sea-ice extents. Two simulations used observed monthly-mean SSTs corresponding to the years 1980 through 1990. The other two used climatologically-averaged monthly-mean SSTs, which were obtained by time-averaging observed SSTs for the years 1980 through 1990. Salient features of our simulations are shown in Table 4.

Progress

Evaluation of the multi-year time-averaged climate

The first part of the analysis aimed at evaluating how well the CAM2 model simulates the multi-year time-averaged present global climate at each resolution. The analysis consisted of comparing 22 simulated variables to corresponding observations (or reanalysis), using Taylor diagrams**.** These meteorological quantities, and the observational data sets used, are shown in Table 2. Prior to comparison, monthly-mean climatology was calculated for each variable from the model output and from the observations or reanalysis, by averaging monthly-mean results over all available years. Also each simulated variable was regridded to the grid of the corresponding observation (which varies from observation to observation).

On the Taylor diagram, each simulated meteorological quantity is plotted on a polar coordinate system. The angular coordinate shows the correlation coefficient between the simulated and observed results, calculated over the spatial domain and over the 12 months of the climatology. This measures to what extent the maxima and minima in the model results are in the correct geographical locations; it does not assess whether the magnitude of spatio-temporal variability in the model results is correct. That property is measured by the standard deviation of the simulated variable calculated over the different grid cells and over the 12 months of the climatology, normalized by the standard deviation of the corresponding observed variable calculated in the same way. This is plotted on the radial coordinate of the Taylor diagram. On this diagram results of a perfect model would be plotted on the x-axis with a radial coordinate value of 1; the distance on the plot between this "ideal point" and the point where the model results are plotted represents the RMS error of that variable with respect to its reference observations. Thus, the farther a variable is from the ideal point, the poorer is the agreement between the model simulation and the observation for that variable as measured by the RMS error.

Resolution	Prescribed SSTs	Number of years analyzed
B32 (2 degrees x 2.5 degrees)	Climatology calculated from AMIP [2]	10 years
C32 (1 degree x 1.25 degrees)	AMIP [2]	10 years (1980 through 1990)
D32 (0.5 degree x 0.625 degree)	AMIP [2]	5 years (1980 through 1990)
0.4 degree x 0.5 degree	Climatology calculated from AMIP [2]	5 years

Table 4: Description of the simulations

A Taylor diagram was plotted for results of each CAM2 simulation (Figures 14a through 14d). The zg500, ta850 and tas variables show a very good agreement with observations across the different resolutions. Other variables such as prw, rsut, rsutcs, or hus850 also show a relatively good agreement with the observations. On the other hand, the clt, ta200, tauv, va200, va850, hfls and pr variables agree relatively poorly with observations. It has to be noted though, that the apparent ability of the model to simulate a specific variable may for some variables depend strongly on the reference observation chosen. Different observational data sets sometimes differ significantly from one another, and a variable that has a good agreement with a set of observations may have a poor agreement with another. This can be the case for variables such as hfls, tauu or tauv.

Figure 15 is a Taylor diagram representing all the variables from all the simulations. In order to show how the agreement between the simulated variable and the observation depends on model resolution, the dots corresponding to the different simulations of a same variable were connected by arrows (the arrows go from one resolution to the next finest resolution, for example from 2 x 2.5 to 1 x 1.25). For most variables, the agreement of the model with the observations either does not improve or improves very little from the B32 simulation (2 degrees x 2.5 degrees) to the finest-resolution (0.4 degree x 0.5 degree) simulation. (i.e., the dots representing the different variables generally do not get significantly closer to the point representing the reference observation as the resolution increases.)

The same exercise was performed for two global climate models that are closely related to CAM2. The CCM3 model is the immediate predecessor to CAM2; we analyzed results of this model obtained using Eulerian spectral dynamics at T42 and T170 truncations (resolutions of 2.8 deg and 0.7 deg, respectively). As configured for our simulations, CAM2 differs from in both dymanics and aspects of the model physics. The CCM3 results are shown in Figure 16a. These results show greater sensitivity to spatial resolution than results of CAM2 do. We also analyzed results of the NASA/NCAR FV General Circulation Model (GCM; http://gmao.gsfc.nasa.gov/sci_research/NASanCAR/). We ran this model at the B32 and D32 resolutions (2.0 x 2.5 deg, and 0.5 x 0.625 deg, respectively.) This model uses the same physics as CCM3, and the same FV dynamical core as we used in CAM2. The FVGCM results are shown in Figure 16b. In the FVGCM results, most variables agree better with the observation in the higher resolution simulation (most arrows point toward the point of reference). Thus, the FVGCM results show more sensitivity to spatial resolution than both the CAM2 results and the CCM3 results.

Simulated response to 1997-1998 El Nino

We assessed how well the CAM2 model simulates the effects of El Nino on climate at model resolutions of 2 degrees x 2.5 degrees and 0.4 degree x 0.5 degrees. We did not predict the occurrence of El Nino, but rather its climatic consequences (precipitation anomalies, etc.) At both B32 and D32 resolutions, we performed ensembles of 5 simulations forced with prescribed observed sea surface temperatures (SSTs) from 1997-1998, when the most recent, largest, and best-observed El Nino occurred. For comparison, we also performed ensembles of simulations forced by muti-year average (climatological) monthly-mean SSTs.

At the global scale the model is able to simulate the main spatial patterns of the DJFM El Nino precipitation anomaly (not shown). However, the details of these patterns are often misrepresented.

At the 2 degrees x 2.5 degrees resolution (Figure 17), the CAM2 model simulates the general pattern of El Nino induced precipitation anomalies in the U.S.; however, these anomalies are generally too weak (Figure 18ab). At 0.4 degree x 0.5 degrees (Figure 17c), the agreement between the model and the observations seems to get worse. In particular, the precipitation increase over the Pacific Ocean and over the west coast of the U.S. is almost totally absent. This seems to be due to an overestimation by the model of the climatological precipitation over these regions.

As for precipitation, the CAM2 correctly simulates the gross spatial patterns of DJFM El Nino-induced anomalies of skin temperature, sea level pressure, and the eastward wind speeds at 20,000 Pa and 85,000 Pa. However, the regional details of these anomalies are often poorly represented. None of these simulated anomalies showed any obvious improvement when the CAM2 resolution was increased from 2.0 degrees x 2.5 degrees to 0.4 degree x 0.5 degrees. For some variables, such as precipitation, the model results are worse at the higher resolution. This is generally consistent with the model's weak sensitivity to resolution in simulating the time-averaged present climate.

The El Nino phenomenon has a strong signature on clouds. In particular, it changes the vertical distribution of clouds over certain regions, especially the tropical Pacific Ocean. We evaluated the ability of the model to simulate El-Nino induced changes in vertical distribution of clouds, using the approach of Cess *et al*. [6].

First we plotted vertical profiles of the cumulative cloud frequency, the frequency of clouds found above a given altitude, for both the climatological DJFM season and for the 1997-1998 DJFM (El Nino) season (Figures 18a and 18b).

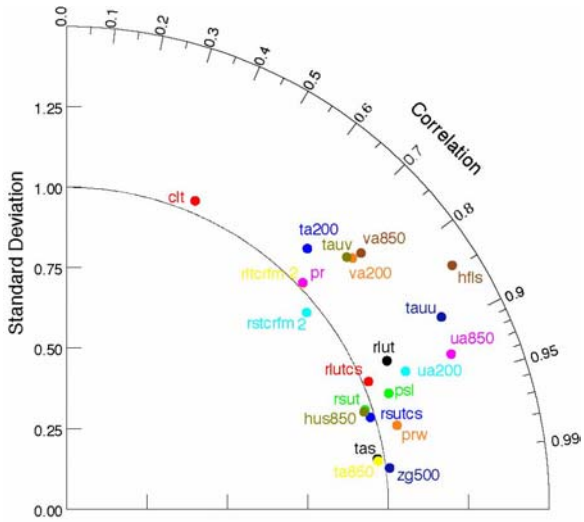
Over the tropical Pacific, the observed cumulative cloud frequency goes from low values at 15 km to values close to 100% (sky totally cloudy), when integrated vertically over the whole column) near the surface, in all cases. For the western Pacific region (Figure 18a), the vertical profile of the observed cumulative cloud frequency strongly shifts to the left from the 1985-1991 DJFMA climatology to the 1997-1998 DJFMA seasonal mean. This indicates that the amount of high clouds decreased during the 1998 El Nino event over the western Pacific, while the amount of low clouds increased. On the contrary, over the eastern Pacific region (Figure 18b), the cumulative cloud frequency remains low above the altitude of 3 km (it reaches the value of 40% at about 3 km) in the climatological DJFMA season, while it is strongly shifted to the right (the value of 40 % is now reached at 13 km) in the 1997-1998 DJFMA season. Similarly, this

indicates that the amount of high clouds dramatically increased over the Eastern Pacific during the 1997-1998 El Nino events, while the amount of low clouds decreased.

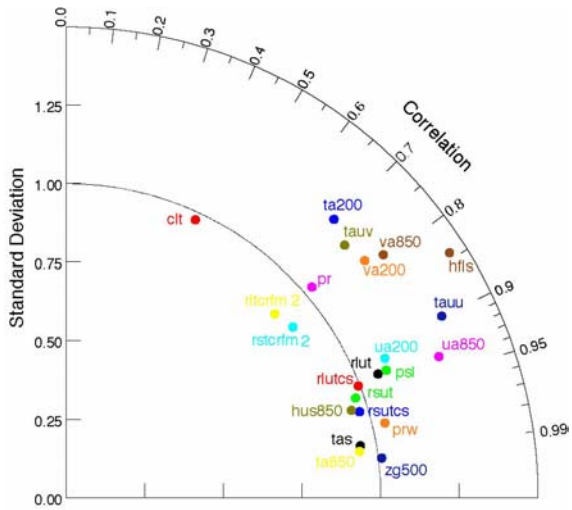
For CAM2, over the Western Pacific (Figure 18a), the vertical profile of the simulated climatological cumulative cloud frequency is in relatively good agreement with observations, as it shows a domination of high clouds at both resolutions. However, in the 1997-1998 DJFM seasons, the vertical profiles of simulated cumulative cloud frequency do not show as large a decrease in high clouds or an increase in low clouds as is seen in the observations. Over the eastern Pacific (Fig. 19b), the difference between the vertical profiles of cumulative cloud frequency in the climatological and the El Nino simulations is greater, but there are still large misrepresentations. We also noticed that the CAM2 model tends to underestimate the total cloud cover, as the cumulative cloud frequency does not reach values close to 100% near the surface.

In summary, although the CAM2 model is able to simulate the large-scale spatial patterns of the climate response to El Nino, the details of this response are often misrepresented. For example, over the western U.S., the positive precipitation anomaly is underestimated at the 2 degrees x 2.5 degrees resolution and is almost nonexistent at 0.4 degree x 0.5 degree. Simulated El Nino-induced changes in the vertical distribution of clouds in the tropical Pacific are qualitatively correct but smaller than observed changes.

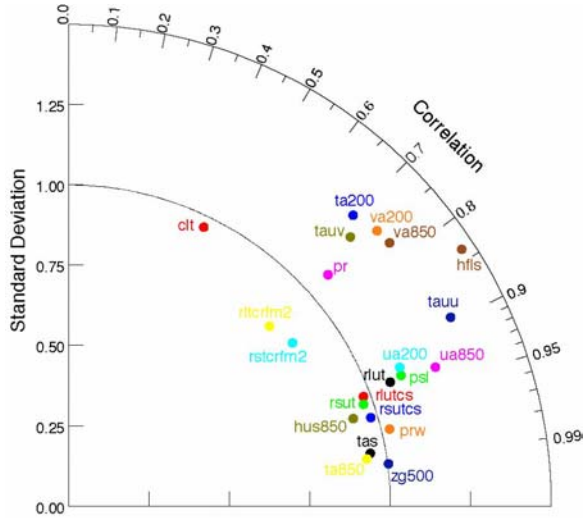
a. B32 (2 degrees x 2.5 degrees)



b. C32 (1 degree x 1.25 degrees)



c. D32 (0.5 degree x 0.625 degree)



d. 0.4 degree x 0.5 degree

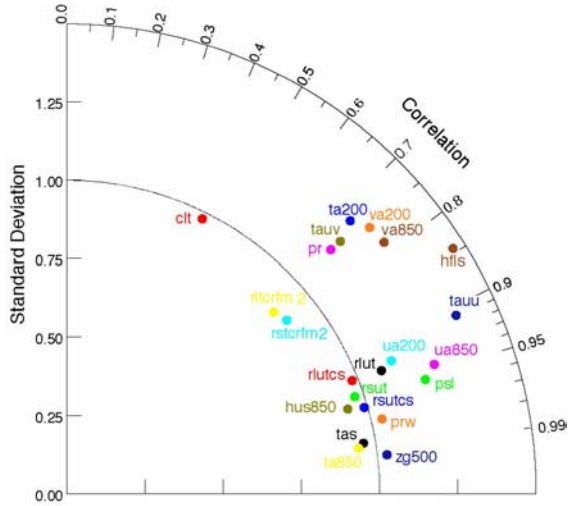


Fig. 14. Taylor diagrams for the CAM2 simulations at four different spatial resolutions. Each diagram shows the agreement (as measured by the RMS error) of 22 simulated variables with the corresponding observation or reanalysis.

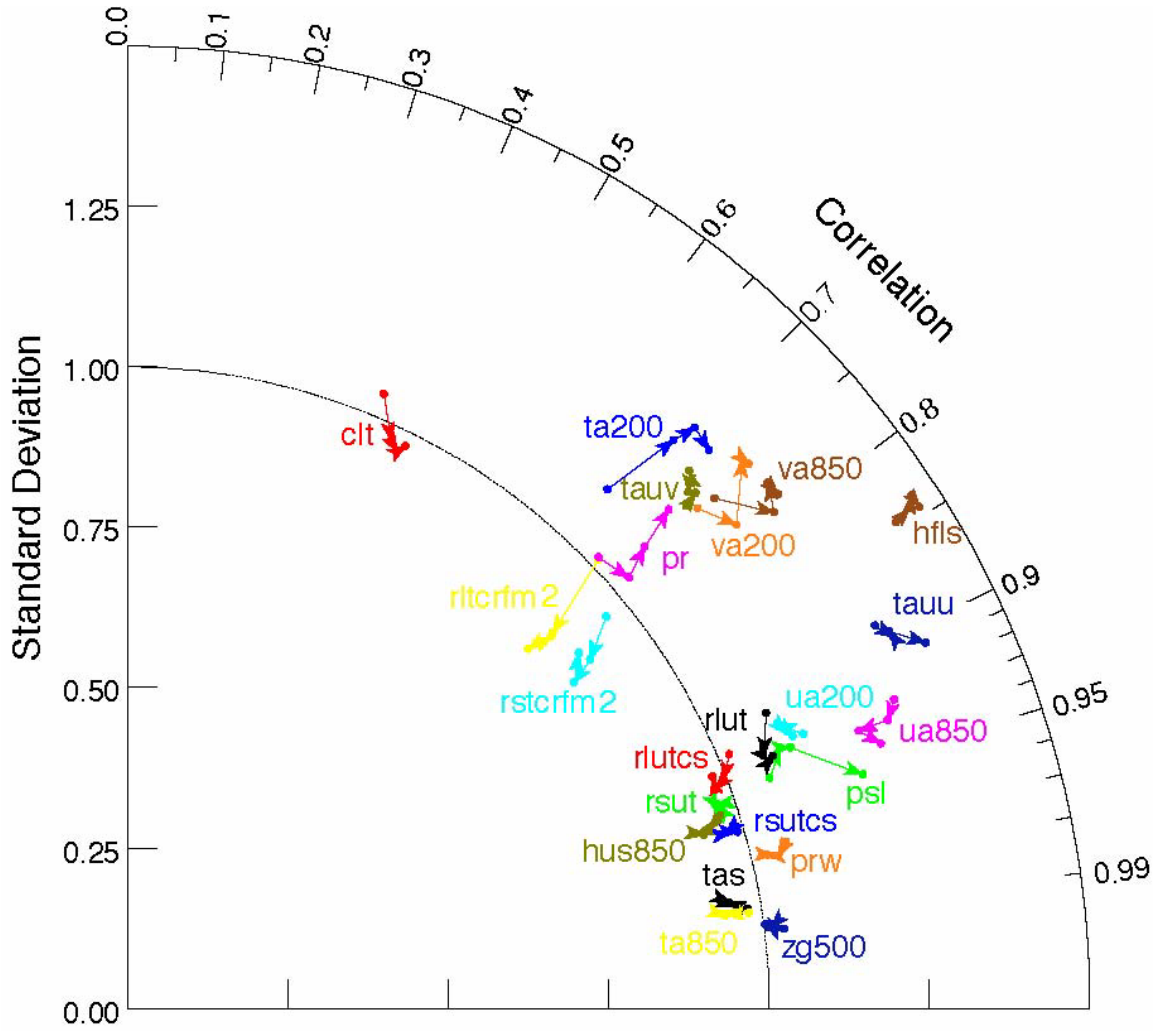
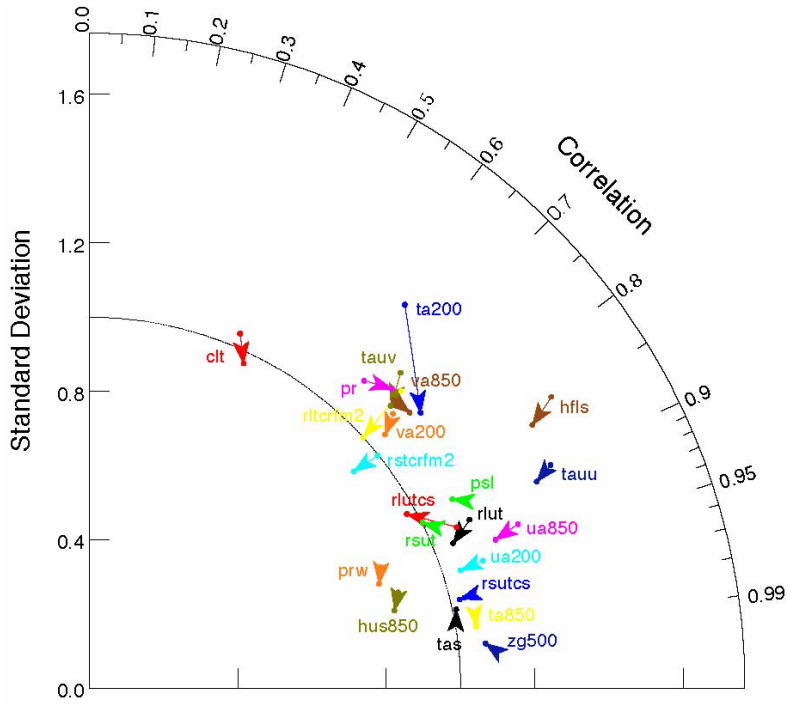


Fig. 15. Taylor diagram representing the CAM2 simulations at all four resolutions. The arrows go successively from B32, to C32, from C32 to D32, and from D32 to 0.4 x 0.5; they show for each variable how the agreement of the simulations with the observations or reanalysis evolves across the resolutions.

a. CCM3 at T170 Vs CCM3 at T42



b. FVGCM at 0.4 degree x 0.5 degree Vs FVGCM at 2 degrees x 2.5 degrees

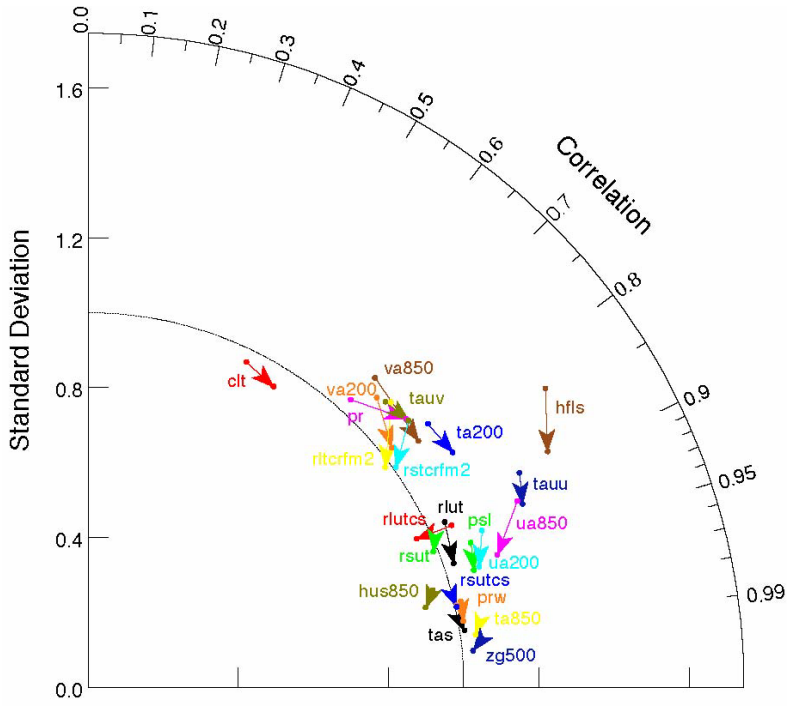
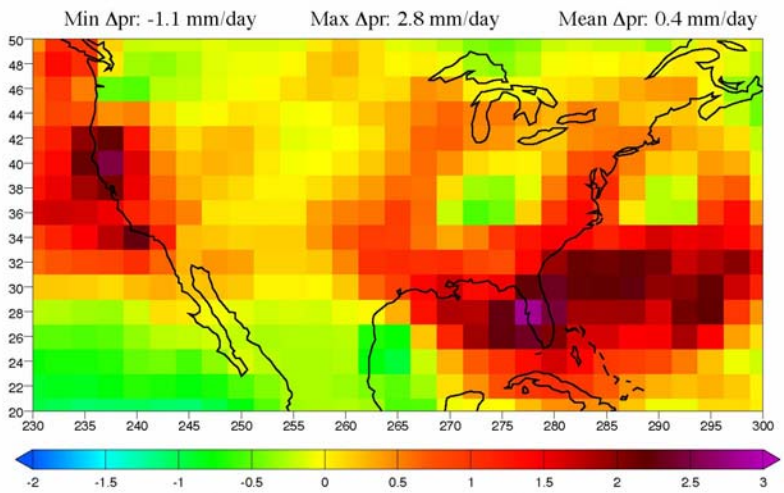
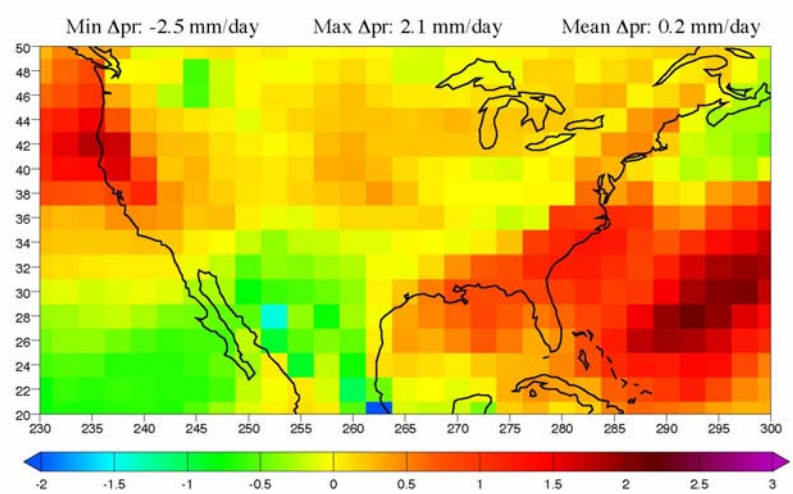


Fig. 16. Taylor diagrams for the CCM3 model ((a) the arrows go from T42 to T170) and for the FVGCM model ((b) the arrows go from 2 x 2.5 to 0.4 x 0.5).

a. CMAP observation



b. 2 degrees x 2.5 degrees



c. 0.4 degree x 0.5 degree regridded to 2 degrees x 2.5 degrees

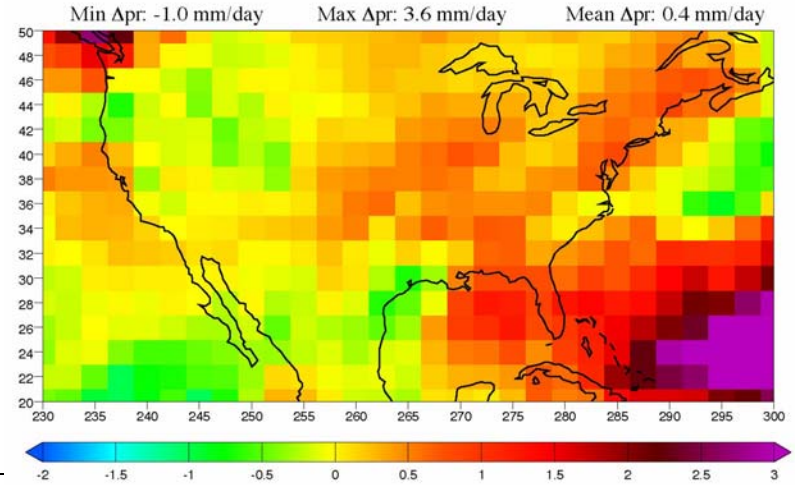
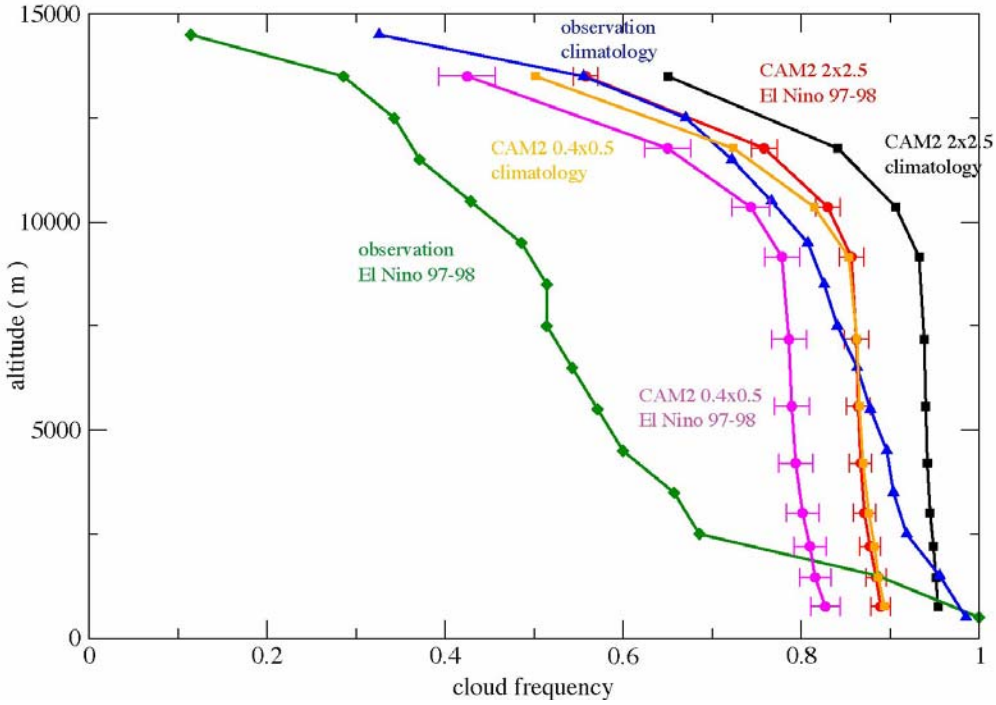


Fig. 17. Precipitation anomaly for the 1997-1998 DJFM, over the continental U.S.

a. Western tropical Pacific



b. Eastern tropical Pacific

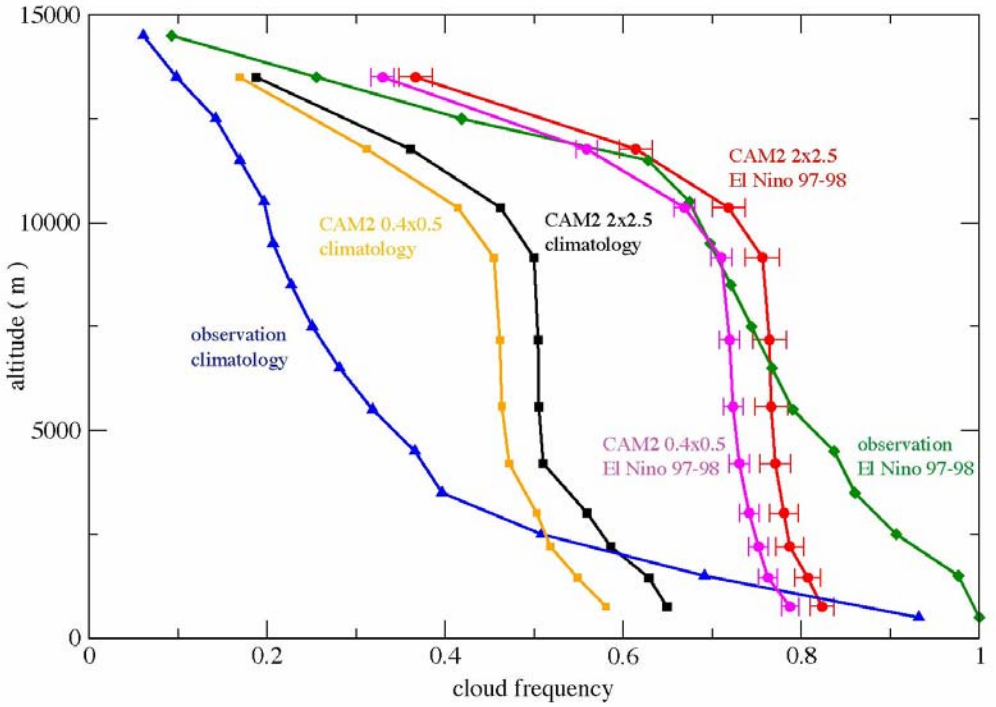


Fig.18. Vertical profiles of the cumulative cloud frequency over the tropical Pacific. The observational data are provided by Cess et al.

5.6 PARALLEL I/O FOR CAM (W. YANG AND C. DING)

Multiple Precisions for History Buffers

In CAM history netCDF I/O, history information are stored in real*8 buffers, but are often written to output files in real*4 precisions. This scheme has two drawbacks: (a) it allocates more memory than necessary, (b) the internal data type conversions inside netCDF slow down data transfer significantly (see Figure 19).

To improve efficiency and also reduce memory, we have implemented multiple precision capabilities for history buffers in CAM controlled by a runtime parameter. Given a specific precision requirement, the minimum amount of memory is allocated and conversions between different processors are performed outside of netCDF.

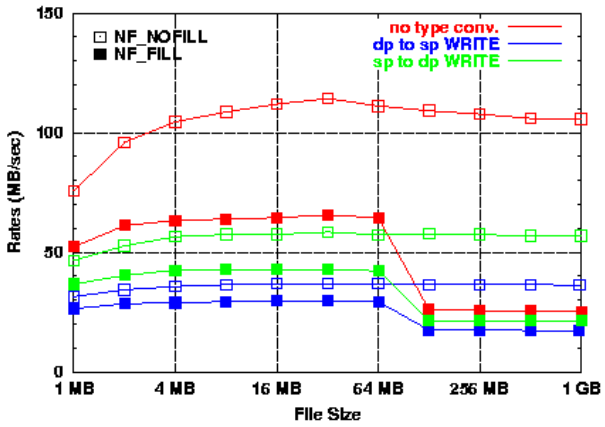


Fig. 19. Effect of netCDF internal data type conversion. Three types of write rates are shown: (1) Red: writing single precision buffer array to disk file in single precision; (2) blue: writing double precision buffer array to disk file in single precision; and (3) green: writing single precision buffer array to disk file in double precision. Clearly, internal precision conversion inside netCDF slow down data writing rates by 30-50%.

Parallel I/O with ZioLib

When introducing parallel I/O ability into CAM while making the user interface simple, we had to consider a few important issues. First, we note that a distributed array is read or written at a maximum rate when the parallel decomposition is made along the last array dimension ("Z-decomposition"). Since this is not the case for arrays in CAM, remapping arrays into a Z-decomposition is more important in order to increase I/O bandwidth than simply making more processors stage I/O. It is essential in this regard to have a remapping scheme that will work for any distributed array of an arbitrary parallel decomposition. Second, some fields are written to a disk file in a different array index order than the one in CPU resident memory. The I/O scheme to be implemented in CAM, therefore, should allow writing the global field in a different index order than the one used in local array, and this must be done in parallel. Third, the I/O scheme must provide a simple and unified user interface that will work for all dynamics decompositions (i.e., 1- and 2-D parallel decompositions) and all physics decompositions (i.e., different physics "chunking" options) adopted in CAM.

We have developed a software library, 'ZioLib' to accommodate such needs. The library remaps distributed arrays into Z-decomposed arrays on a subset of processors, and from there, launches a write to a disk file (see the Fig. 21, below). In this Z-decomposition, the data layout of the remapped arrays on the staging processors' memory is the same as on the disk, thus only block data transfer occurs without multiple seeks during parallel I/O, achieving maximum efficiency.

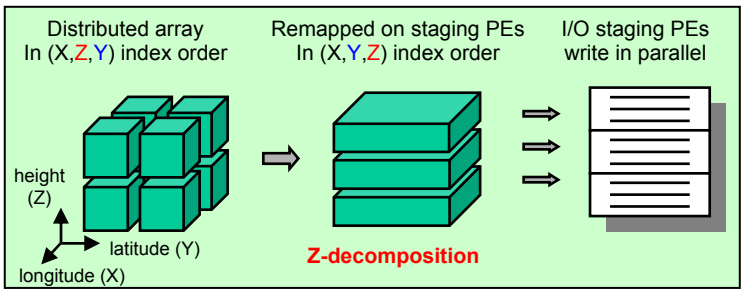


Fig. 20. Parallel decompositions of ZioLib.

ZioLib is a set of Fortran 90 modules supporting netCDF and Fortran unformatted I/O for a distributed array of any block parallel decomposition in any array index order. To provide uniform parallel I/O interface for CAM's different physics "chunking" options including the one where local data consists of multiple disjoint rectangular XYZ-blocks, ZioLib has capability to read and write the global field of a distributed array when local data is represented by multiple rectangular blocks. We consider this feature important because all the parallel netCDF libraries developed so far require that a single rectangular block represent the local subdomain. ZioLib bridges between such parallel libraries and CAM, and still performs parallel I/O.

ZioLib supports four-byte integer, four- and eight-byte real data types of an arbitrary number of array dimensions. Because a global array is gathered on a few staging processors instead of a single root processor as in the case of the current CAM, memory limitation on each processor is eliminated with ZioLib. Since only the staging processors, not all computing processors, actually access a disk file, I/O server nodes will be congested less. ZioLib yields robust I/O performance regardless of original parallel decomposition types. As ZioLib performs all necessary remapping tasks, an explicit gather/scatter of a global array for I/O will be unnecessary in CAM. Details about the algorithm, performance data, source codes, test codes and information on how to implement in CAM are available on our webpage. http://www.nersc.gov/research/SCG/acpi/ZioLib.

We have tested ZioLib for various parallel decompositions using a 3D array of 256x256x256 (128 MB) in (X,Z,Y) order, to be written out in Fortran binary mode to a file in (X,Y,Z) order. Compared to the existing CAM single-processor

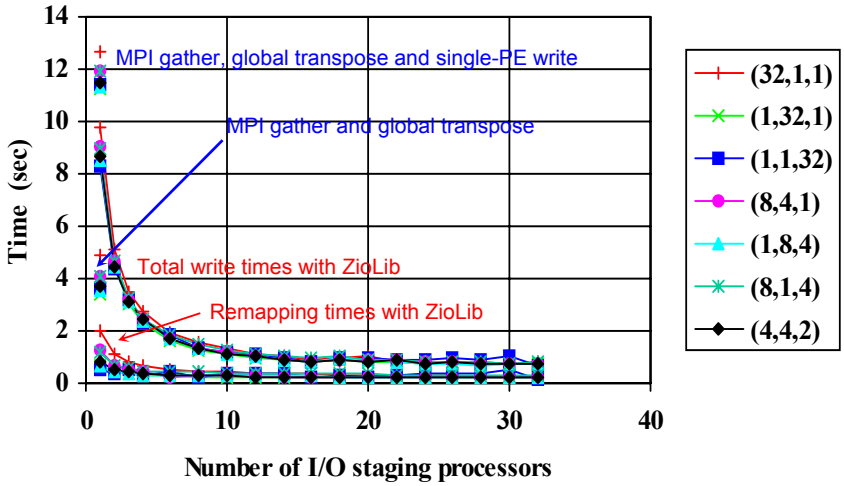


Fig. 21. Parallel Performance of ZioLib of a 256^3 array in 7 different domain decompositions.

I/O method, where a single root processor gathers a global field from distributed arrays, transposes it into the correct output array index order, and then writes to a disk file, ZioLib speeds up writes by a factor of up to 17 on 32 processors (see the Fig. 22, above, in the figure, (8,4,1), for example, refers to a parallel domain decomposition with eight, four and one processors along the X-, Y- and Z-axes, respectively). It speeds up reads by a factor of up to 23. Note that ZioLib yields almost uniform I/O rates regardless of decomposition types. Repeating the same experiment with the CAM's T42L26 resolution, we observe that ZioLib can speed up unformatted writes by a factor of 3-4 and reads by a factor of 6-7.

We have completed a full implementation of ZioLib into the Eulerian dynamic core version of CAM for all netCDF and Fortran unformatted input and output, and confirmed, using the T42L26 resolution on the IBM SP at NERSC, bit-for-bit agreement of the generated history and restart output files with those generated with the existing CAM. We have also confirmed bit-for-bit agreement with and without the load-balancing physics "chunking" option. Even with the serial netCDF library, ZioLib can speed up history I/O by a factor of 1.5-2.5 in comparison to the existing CAM I/O method, thanks to a more efficient way of gathering global fields.

The implementation results in considerably simpler coding for I/O by eliminating highly complex segments for gathering, scattering or transposing global fields for various data decompositions and different dynamic cores. We put a unified I/O approach in place by providing a single set of netCDF wrapper routines and a single set of unformatted I/O wrapper routines and sticking to them throughout CAM, instead of allowing various I/O schemes to proliferate as in the case of the current CAM. Most importantly, the modifications have introduced parallel I/O ability into CAM. For that, we provide an easy way in switching between serial and parallel I/O – just the numbers of I/O staging processes need to be set in a single module to whatever numbers appropriate on a particular computer platform. The next phase must be to complete a ZioLib implementation in Community Land Model (CLM), which is closely related to CAM and shares many routines.

Parallel NetCDF Libraries

We are working with Majdi Baddourah of NERSC on developing a parallel netCDF library. This library has both C and Fortran interfaces, and supports most of the functions of the serial netCDF library. The beta version is now for public use on the NERSC IBM SP. Stacy Walters of NCAR showed interest in testing this parallel netCDF library on the NCAR's IBM SP.

As part of the development endeavor, we have written a comprehensive parallel netCDF I/O test code in both Fortran and C, which can test for consistency, I/O rates, different parallel decompositions, different grid sizes, etc. Write rates of a 128 MB array distributed in a Z-decomposition are about 170 and 160 MB/sec on 8 and 16 processors, respectively. Read rates are about 330 MB/sec on both 8 and 16 processors.

A parallel netCDF library is being developed at Northwestern University and Argonne National Laboratory under the Scientific Data Management project of SciDAC for use in the distributed memory computers. We provided a comprehensive parallel netCDF I/O code that tests arbitrary parallel domain decomposition on a 3-D array. The test code is part of the standard test for the NU/ANL parallel netCDF releases (see acknowledgement of their SC'03 paper [7]). We have also tested the NU/ANL codes and sent feedback on our results to them. We are studying the interface and plan to integrate it into our ZioLib for implementation into CAM and other CCSM components.

.

6.0 OCEAN MODELING (TOPIC COORDINATOR: P. JONES)

Background

Prior to the SciDAC program, development and validation of ocean models was done under the aegis of the CHAMMP program and the CCPP. POP was developed, tested, validated and released under those programs. Other work focused on the Miami Isopycnic Coordinate Ocean Model (MICOM), which was brought to LANL by its creator, Rainer Bleck. Comparisons between POP and MICOM were made that revealed strengths and weaknesses of each. This led LANL to pursue the development of hybrid-vertical coordinate models on the premise that they might combine the advantages of POP in the surface mixed layer with those of MICOM below the thermocline. Two more models have arisen from that working hypothesis: the HYCOM model derived from MICOM and the HYPOP model derived from POP. By building two models from different roots we can evaluate quantitatively the extent to which they yield the same climate.

6.1 PARALLEL OCEAN PROGRAM: POP 2.0

POP Objectives

The objectives for ocean modeling outlined in the original proposal pertained to the POP and HYPOP models. POP is based on traditional fixed-z-level coordinates. It is a mature code with a large user community. The NCAR CCSM2 uses POP as the ocean component. Also the Community Ocean Biogeochemistry Model that is being developed with support from the National Ocean Partnership Program (NOPP) is based on POP.

Table 5. POP objectives and tasks	**Status**	**Deliverable**	**Date**
Framework compatibility			
CCSM utility layer	Done	POP 2.0	
Machine-specific layer	Done	POP 2.0	
Performance optimization			
Cache-friendly decomposition	Done	POP 2.0	4Q02
Static load balancing	Done	POP 2.0	4Q02
Adaptation to new hardware	Ongoing		
Hybrid MPI/OpenMP	Done	POP 2.0	3Q02
Vectorize for Earth Sim & Cray X1	Done	Vectorized code	2Q03
MLP	Installed	Experimental code	1Q03
Dynamic load balancing scheme	Not needed	N/A	N/A

The SciDAC objectives regarding POP were two-fold: (1) Apply modern software design methods to POP by gradually modify POP so that it becomes "framework compatible." This means that it should be designed to interface to a "CCSM utility layer" and a "machine-specific" layer. Ultimately these layers will be part of the ESMF. (2) Improve the parallel performance and scaling of POP while at the same time increasing its portability across a range of architectures. The details of these objectives, their status, deliverables, and delivery dates are summarized in Table 5.

POP Progress

As part of an ongoing process designed to arrive at a version of POP that is based on ESMF when it becomes available, POP has been modified to make use of software layers that support machine-dependent and utility routines.

POP was originally developed for the Connection Machine, a massively parallel computer built with vector processors (which did not use cache) running in single instruction, multiple data (SIMD) mode. It was later modified to work on multiple instruction, multiple data (MIMD) parallel computers built with commodity scalar processors. Such processors typically have poor memory bandwidth, which is at least partially offset by the use of cache. However, if the "chunks" into which the work is divided are too large to fit in cache, many cache misses will occur even while doing a single "chunk." POP 2.0 contains the capability to subdivide work into chunks small enough that each will fit in the cache; this is a "cache-friendly" decomposition. This finer-scale decomposition provides a larger number of smaller pieces of work, which makes easier both static load balancing and a mixed coarse-grained-MPI with fine-grained-OpenMP programming style. Adaptation to new (and old) computer architectures is an ongoing task. After the demise of the CM-5, vectorization went out of vogue, but was recently reincorporated in POP for use on the Japanese ES and the Cray X1. MLP, invented by Jim Taft (NASA Ames), can be an effective replacement for MPI on parallel machines that support some form of globally shared memory. MLP has been implemented in an experimental version of POP; it gives impressive scaling and speedup.

6.2 HYBRID VERTICAL COORDINATE POP

HYPOP Objectives

The fundamental objective that motivates the development of HYPOP is to capture in a single model the advantages of z-level coordinates (like POP) in the surface mixed layer with the advantages of isopycnic-layer coordinates (like MICOM) in the density-stratified ocean below the thermocline. This overarching objective remains unchanged, but the sub-objectives and tasks given in the original proposal have been revised extensively to clarify the actual steps in the development process. These are listed in Table 5 along with their status (degree of completion or estimated starting date), deliverables, and estimated completion dates.

It was decided early in the HYPOP project to use modern software engineering practices in the development of HYPOP. A Requirements Document and an Architecture Design Document have been completed in draft form, but it is to be expected that both will evolve as new requirements are perceived and new generations of computers are invented. Like POP, HYPOP will be constructed to make use of the ESMF software and standards as they become available. Therefore, HYPOP uses the same software-layer structure as POP. Table 6 contains a detailed roadmap of the work that has been completed, is in progress, and targeted for the future.

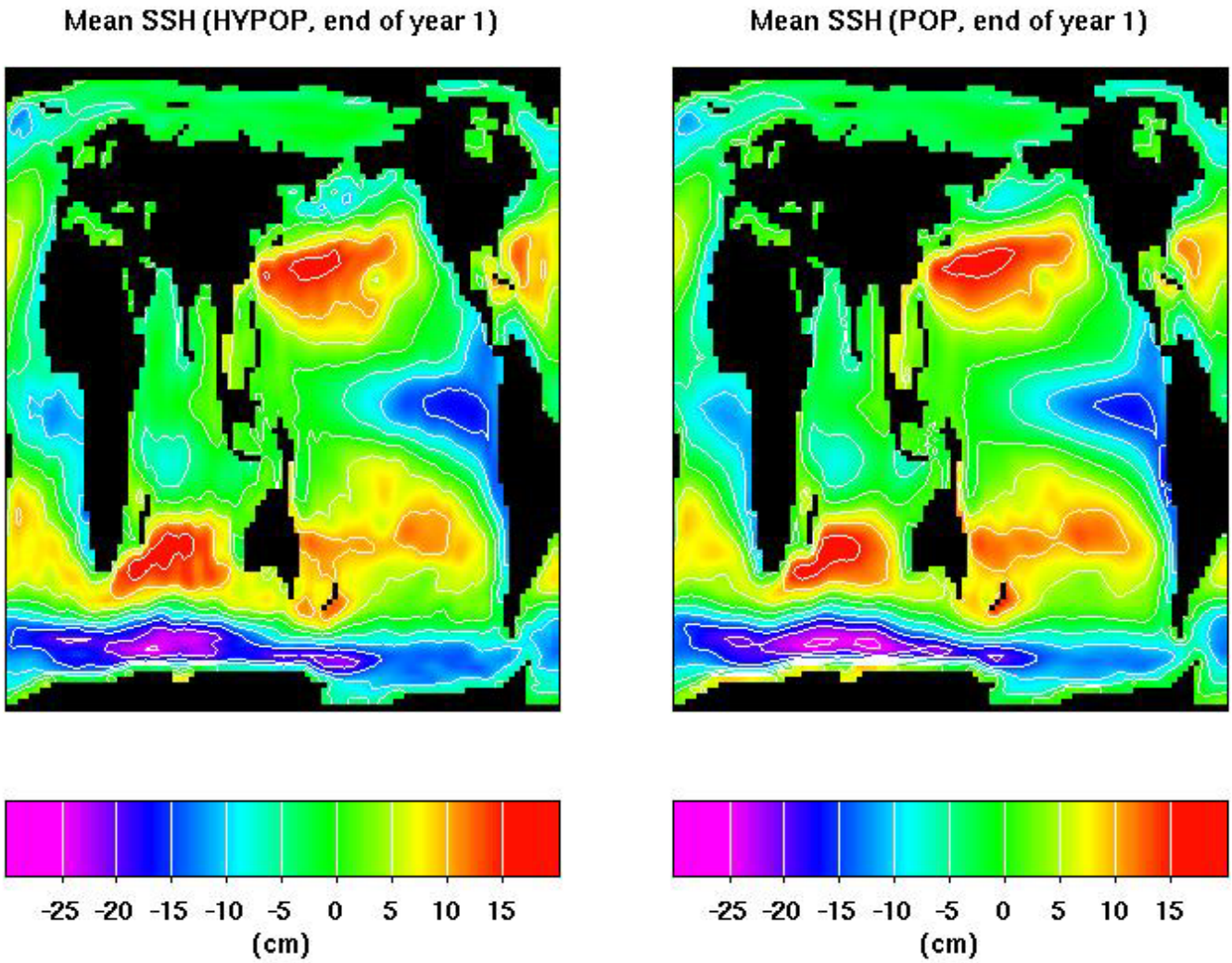
Table 6. HYPOP objectives and tasks	**Status**	**Deliverable**	**Date**
Software design and documentation			
Requirements document	Done	Document	
Architecture design document	Done	Document	

Framework compatibility			
CCSM utility layer	Done		
Machine-specific layer	Done		
New dynamical core			
Develop framework for hybrid ALE model	Done		
Develop horizontal remapping advection	Done	Publication	
Develop vertical-remapping ALE scheme	Done		
Resolve B- versus C-grid issue	Done		
Treatment of Coriolis terms on C-grid	Done		
Develop Laplacian friction parameterization	Done	Publication	
Implement Jackett-McDougal EOS	Done		
2D (SWE) tests: flat bottom and bowl topography	Done		
Multi-layer dynamics			
Define orthobaric density variable	Done		
Develop improved barotropic-baroclinic splitting	Done	Publication?	
Develop Boussinesq correction algorithm	Done	Publication?	
Implement correction in momentum equations	In progress		4Q03
Adapt horizontal remapping to layer model	In progress		4Q03
Adapt KPP parameterization to layer model	In progress		1Q04
Resolve issues with layer-topography interaction	In progress		1Q04
Specify and test hybrid vertical grid scheme	In progress		2Q04
Develop sub-grid-scale viscosity and diffusion	4Q03		2Q04
Multi-layer tests with full (Eulerian) vertical remapping to compare with POP			
Full topography tests, compared with POP	In progress		3Q03
Suite of test problems, pure Eulerian mode vs POP	3Q03		2Q04
Suite of test problems, hybrid mode vs HYCOM	1Q04		3Q04
Parallelization and user interface			
Add initialization, forcing, diagnostics from POP	4Q03		2Q04
Parallelize code (MPI, OpenMP, then MLP)	4Q03		2Q04
Add sub-blocking and F90 features from POP	2Q04	Parallel code	4Q04
Tests for tracer advection; comparisons with observations and other models			
Stand-alone simulation tests	2Q04		4Q04
Tracer validation tests	2Q04	Biogeochem	1Q05
Coupled model tests	1Q05	Climate	3Q05

HYPOP Progress

One of the guiding principles of the original SciDAC proposal was to focus on the "dynamical cores" of the component models of the CCSM. The reason for this is that the dynamical core is usually the most computationally demanding part. It also strongly influences the data structures around which the rest of the model is built. Therefore, enhancement of model performance entails optimization of the dynamical core. As is apparent from Table 6, HYPOP is progressing well. We will not repeat material presented in past progress reports, but will focus on a recent important breakthrough in the separation of the barotropic and baroclinic modes in HYPOP.
The computational Achilles' heel of POP is the implicit treatment of the barotropic mode. A preconditioned conjugate-gradient-residual solution technique is used; it requires two global

reductions per iteration. This becomes a burden on parallel machines with high-latency (>5 μs) networks, and eventually limits scaling to high processor-count even on low-latency (< 1 μs) machines like the SGI Origin 3000. A major design requirement for HYPOP is that an explicit subcycling method be used to solve the barotropic mode.



Several attempts to develop a method in HYPOP for splitting the barotropic mode from the baroclinic mode failed to produce a robust method that worked in the presence of realistic bottom topography. This past spring, John Dukowicz and Rick Smith developed a new method based on the solution of the vertically integrated momentum and mass continuity equations. These equations are solved explicitly by subcycling with a short barotropic timestep and employ an accurate and efficient remapping scheme for the transport of column mass. A unique feature of the algorithm, which distinguishes it from all other barotropic subcycling methods currently used in Oceanic General Circulation Model (OGCMs), is that parameters can be tuned so that the high-frequency, high-wavenumber, gravity-wave modes can be retained if desired. Other methods damp these modes to prevent numerical instability. The method has been implemented in HYPOP and validated by comparison with similarly configured POP runs.

Plans

Features added to POP 2.0 include partial bottom cells, a tripole grid, and sub-blocking for cache-based machines. More recently, POP has been vectorized for the Japanese ES and the Cray X1. Except for the need to respond to new developments in computer architectures such as "floating-point gate arrays," development of POP is fairly complete. Because POP has such a large user community, we expect it to be applied to a wide variety of problems and to have a 5-10 year lifetime, jointly supported by LANL as POP and by NCAR as the Community Ocean Model (COM2).

6.3 GENERIC GRID GENERATOR (W. VOIT, J. DAVIS, R. SMITH)

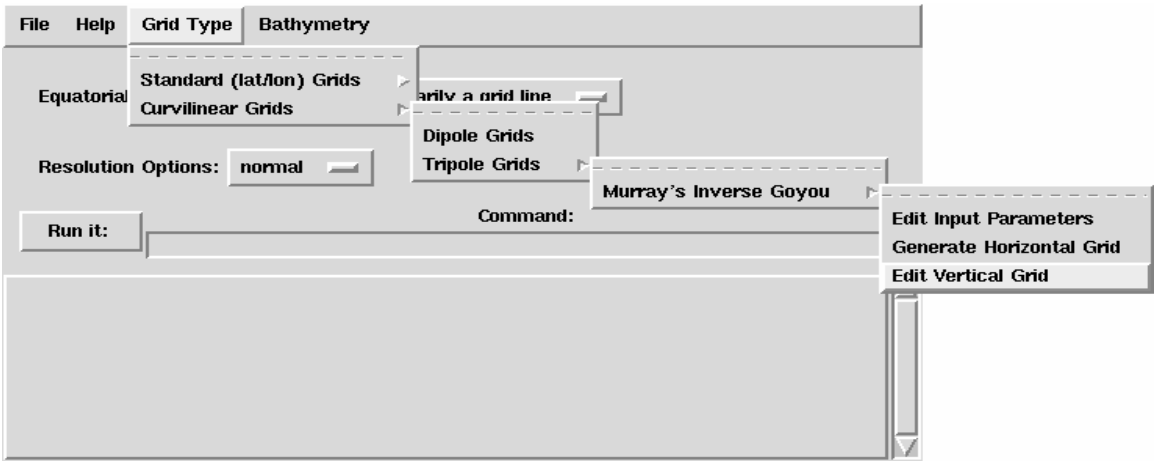
[This section appeared in the previous progress report but is being retained for the benefit of the review committee members.]

The displaced-pole grid (or "dipole grid"), in which the singularity in latitude-longitude grids at the North Pole is moved into an adjacent landmass, has become one of the most widely used features of POP. It has also been implemented in CICE. Murry created another grid, called the "tripole grid" that provides the same benefits as the displaced-pole grid, but is superior in that the grid cell size in the Arctic Ocean varies much less in the tripole grid. The tripole grid has now been implemented in POP 2.0. Both grid types are particularly attractive to paleoclimate modelers, because in past eons, continental drift often left no land masses at either rotational pole. However, it is a laborious process to compute the locations of the grid points and lengths of the cell sides in these non-standard grids. Thus, an interactive tool was needed to help automate the generation of grids at arbitrarily selected resolutions and pole positions.

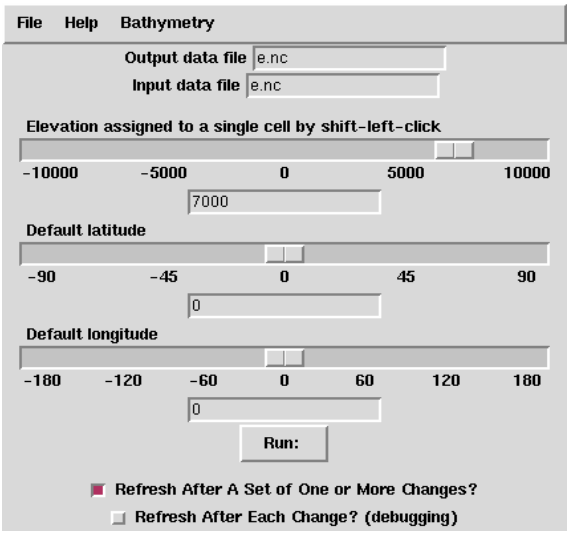
An additional serious difficulty in setting up ocean simulations of the present-day ocean is that even the best datasets on ocean bottom topography contain subtle errors, particularly in the depths and widths of critical sills on the ocean floor. These errors must be remedied by manually adjusting the ocean bottom depth at a series of about 100 locations on the ocean floor, using more accurate values obtained from detailed localized sources. Again an interactive tool was needed that uses point-and-click methods to locate grid points, reports their depth values, and allows the user to modify them.

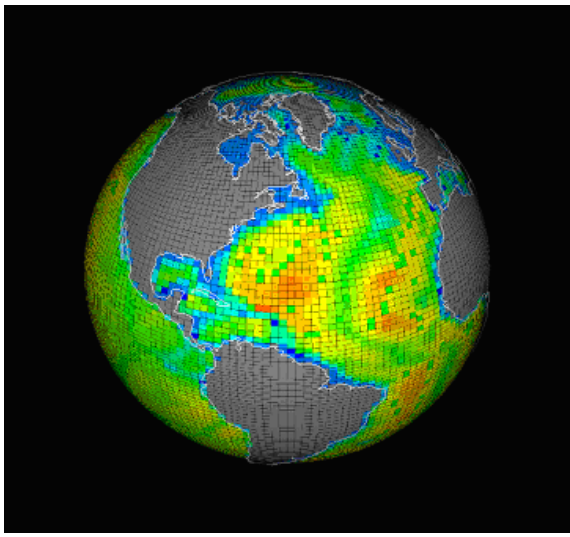
With these needs in mind, just such a tool has been developed. gCubed, the Generic Grid Generator, is an interactive software tool framework for producing computational grids for climate modeling. It is highly customizable. It can contain grid generators, editors, and converters. Each of its component tools can also be run separately from a command line.

Existing tools can be replaced or updated independently. The framework Graphical User Interface (GUI) is implemented with scripts written in Tcl/Tk (Tool command language/Toolkit). The tools themselves can be any program or script that can be run by typing a command into a console window.



The GUI, below, illustrates one sequence of cascading menus. This sequence, culminating in the "Edit Vertical Grid" selection, brings up a new GUI screen for editing the topography of a previously generated model grid with eCubed, the Earth Elevation Editor.





When the Run button is clicked, an editable, three-dimensional image of the gridded globe appears. This example is a tripole grid developed from the work of Ross Murray. It has two north 'pole' singularities, one in Canada and one in Russia.

gCubed is a cross-platform, adaptable system. Its main use to date has been preparation of custom grids for POP. The first stable version (July 2002) runs on Windows and on Unix/Linux.

However, the current development version is preferred due to its several major enhancements, including an undo capability. It has been tested on several Unix/ Linux platforms, but not yet on Windows. This version, including up-to-date documentation, will be given to friendly users upon request. The current gCubed toolset consists of a tripole grid generator, eCubed, and a couple of data conversion utilities.

7.0 SEA ICE MODELING (TOPIC COORDINATOR: W. LIPSCOMB)

Background

The LANL sea ice model, CICE, has a long history of innovation with regard to the "dynamical core." The once-controversial Elastic-Viscous-Plastic (EVP) rheology is now widely accepted and has been adopted in other sea ice models. CICE has also benefited from other innovations in numerical algorithms, particularly the concept of "remapping." Remapping can be thought of as a conservative generalization of "semi-Lagrange" advection. In the semi-Lagrange approach, the velocity field at times [n-1] and/or [n] is used to trace the trajectory of the center of the target cell backward in time to its location at time [n-1]. The advected quantities are interpolated from their values in cells adjacent to the time [n-1] position and then carried forward along the trajectory to the center of the target cell at time [n]. This procedure has several shortcomings, the most serious of which is that advected tracers are not conserved. Remapping generalizes semi-Lagrange by tracing backward in time the four corners of each cell rather than the centers only. Values on the grid at time [n-1] are then interpolated onto the distorted grid at time [n-1] and carried forward to time [n]. This is conservative because each cell of the distorted grid at time [n-1] is mapped into a single cell of the undistorted grid at time [n]. Both sets of cells exactly cover the entire domain. It has been shown analytically that remapping is "monotone," that is to say that it is guaranteed not to introduce new local maxima and minima in the tracer distribution, as so many other advection algorithms do.

Objectives

The original objectives and tasks needed to achieve those objectives for the CICE model are presented in Table 7. The objectives can be divided into four classes: (1) software design, (2) dynamical core improvements, (3) performance optimization, and (4) parameterization improvements. Many of these mimic objectives and tasks that appear in the POP and HYPOP tables, and therefore require no further explanation here.

The implementation of remapping advection was a high priority, because it has an important advantage computationally. Once the geometric factors are computed for the backward trajectories of the four cell corners, the same factors can be applied to any number of tracer species at a much lower cost than for the first tracer. Thus, the cost to advect N tracers is $C(N) = A*1 + B*(N-1)$, where $B << A$. This is very helpful in the CICE model because there are multiple layers of sea-ice, plus snow on the surface that must be advected in a compatible fashion. Remapping will be very advantageous also in POP when biogeochemistry is being is included: there will be dozens of chemical, biological and ecological species that must be advected in a conservative, monotone way.

Table 7. CICE objectives and tasks	Status	Deliverable	Date
Software design and documentation			
Requirements document	Done	Document	3Q02
Architectural design document	Ongoing	Draft	3Q03
Framework compatibility			
CCSM utility layer	Done		
Machine-specific layer	Done		
Dynamical core			
Remapping advection scheme	Done	Publication	2Q03
Inclusion of metric terms	Done	Publication	
Improve dynamic stability	Done	Implemented	1Q04
Performance optimization			
Adaptation to new hardware (ES, Cray X1)	Ongoing	Vectorized	4Q03
Cache-friendly decomposition like POP	1Q04		3Q04
Static load balancing	1Q04		3Q04
MLP	2Q04		3Q04
Dynamic load balancing	3Q04		4Q04
Improved parameterizations			
Automatic-differentiation derived values	On hold		3Q04

Progress

7.1 SOFTWARE DESIGN AND DOCUMENTATION

Julie Schramm of NCAR, working with William Lipscomb and Phil Jones of LANL, has written a requirements document and a software developer's guide for the CSIM. The requirements document describes scientific and software engineering standards for CSIM; it also specifies machine architectures, grids, I/O formats, and forcing/initialization modes that the model should support. The developer's guide provides detailed coding standards and outlines the procedure for adding new code. Although both documents are fairly complete, they are still awaiting official review.

7.2 PERFORMANCE OPTIMIZATION

Work to date has focused on rewriting CICE and CSIM to run efficiently on vector platforms, in particular the Cray X1 and the Japanese ES. Tests on the ES in Fall 2002 showed that several modules of CSIM and CICE, especially the thermodynamic column physics, vectorized poorly. During the past year Clifford Chen of Fujitsu and William Lipscomb of LANL have rewritten CICE with the goal of reducing its vector cost without degrading scalar performance. Vector benchmarking has been carried out on the Fujitsu VPP5000 using a standalone version of CICE.

The first major phase of this project was completed in August 2003. By use of modified loops and data structures, indirect addressing, and simple compiler directives, vector performance of CICE has improved by a factor of eight, while scalar performance is nearly unchanged. The vector-friendly code is now being tested on the ES and Cray X1. Julie Schramm at NCAR has begun porting vector-friendly CICE modules to CSIM.

7.3 DYNAMICAL CORE

The dynamical core of the sea ice model consists of an EVP scheme for solving the momentum equation, a transport scheme for moving ice horizontally, and a ridging scheme that computes the ice strength and converts thin ice to thick ice under deformation. Each of these components has been improved during SciDAC.

Elizabeth Hunke and John Dukowicz improved the EVP scheme by incorporating metric terms to account for curvature on a sphere. In the process they developed a more accurate representation of the velocity and internal stress fields. A paper by Hunke and Dukowicz, *The elastic-viscous-plastic sea ice dynamics model in general curvilinear coordinates on a sphere—Incorporation of metric terms* was published in 2002 in *Monthly Weather Review*.

William Lipscomb and Elizabeth Hunke, with guidance from John Dukowicz, developed an incremental remapping scheme for horizontal sea ice transport. This scheme is conservative, monotonicity-preserving, second-order accurate (except where limited locally to preserve monotonicity), and computationally efficient for large numbers of tracers. It was implemented in CICE in late 2001, made more accurate and efficient during 2002, and added to CSIM in 2003 after careful testing by Julie Schramm. Incremental remapping is more than twice as fast as the previous MPDATA scheme, leading to better than a 20% reduction in total computational time for CICE and CSIM. Remapping also is more numerically robust than MPDATA. A paper by Lipscomb and Hunke, *Modeling sea ice transport using incremental remapping*, has been accepted by *Monthly Weather Review*.

During 2002 and 2003, researchers at the Naval Postgraduate School, Stennis Naval Research Lab, and British Hadley Center found that CICE occasionally crashed in regions of high grid resolution. Investigations by Lipscomb and Hunke traced the problem to a numerical instability caused by interactions between the EVP dynamics and the ice strength on the B-grid. This instability leads to unphysically large velocities and thick ice. Lipscomb modified the ridging and ice strength parameterizations so as to damp the instability without significantly changing model results in stable regions. As a result, CICE is now more robust for high-resolution runs. These changes will soon be tested in CSIM.

7.4 IMPROVED PARAMETERIZATIONS

Jong Kim of ANL performed a sensitivity analysis and parameter tuning study of CICE using automatic differentiation (AD). AD is a robust, quantitative method for choosing parameters that will optimize agreement between models and observed data. Kim found that sea ice thickness is especially sensitive to ice albedo constants, ice and snow density and emissivity, and the prescribed ice salinity profile. He has summarized his results in a draft manuscript.

Plans

Software design and documentation, improvements to the dynamical core, and improved parameterizations are nearly complete. The main tasks remaining in these areas are formal

review of the requirements documents and developer's guide; submission of a manuscript describing Jong Kim's AD sensitivity study; and possibly a publication describing the changes that improve the model's numerical stability.

Work in the months ahead will focus on performance optimization. We plan to improve CICE and CSIM with several features already implemented in POP 2.0. First, we will modify the boundary routines so that the sea ice model can run on a tripole grid. This grid has two poles in Northern Hemisphere continents far from the geographic North Pole, giving a more uniform grid in the Arctic that should improve physical realism and allow a longer time step. Next we will modify the date structures by adding a third horizontal index, allowing the computational work to be decomposed into small cache-friendly chunks (or larger vector-friendly chunks, if desired). This chunking will result in more flexible and efficient static load balancing. Later we plan to introduce dynamic load balancing, which would allow work to be shifted between hemispheres as the ice advances in one hemisphere and retreats in the other. Jim Taft plans to incorporate MLP in CICE for better performance on machines with globally shared memory.

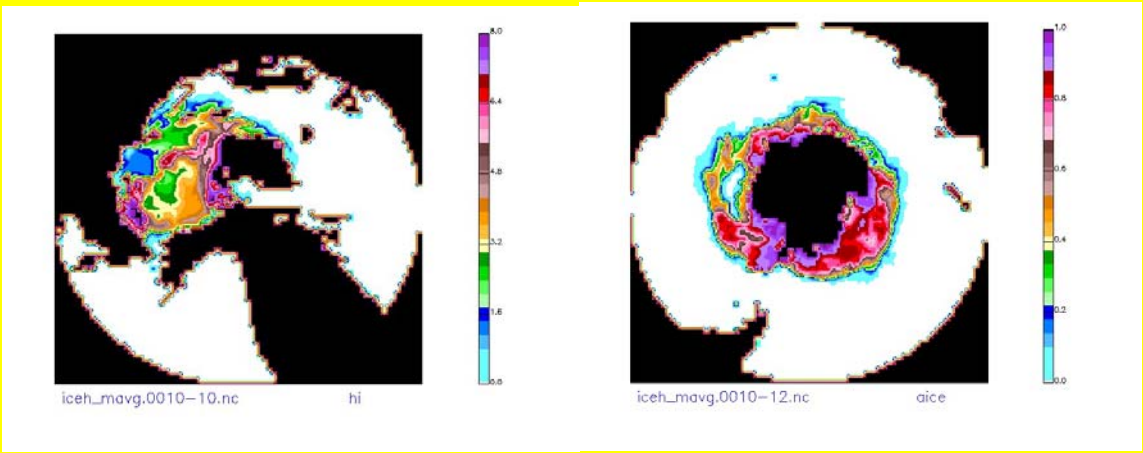
In addition, Julie Schramm will be working with Clifford Chen, William Lipscomb, and NCAR's Tony Craig to produce a vector-friendly version of CSIM that gives the same climate as the current version. For many modules this task is straightforward, since CICE and CSIM already have the same dynamical core and much of the same physics. Some modules, however, have physics differences that could lead to climate differences, requiring further code changes. As a result of this effort, CICE and CSIM will be much more similar than they have been in the past, reducing duplication of labor for code developers.

Sea Ice Coupling with the Parallel Ocean Program

Elizabeth Hunke, Los Alamos National Laboratory

Shown here are simulation results from a global 0.4-degree, coupled ice-ocean (N pole in N. America) POP1.4.3 and CICE. A 10-year atmospheric data cycle (years 1979-1988) was used with NCEP air temperature, humidity, density, and winds at 6-hour intervals to force the ocean and ice. MSU-Xie-Arkin precipitation (monthly), ISCCP shortwave, cloud fraction (monthly) T62 data were read in and interpolated to ocean grid on the fly.

M. Maltrud compiled the monthly river runoff on the ocean grid. For the simulation the ocean took a 32-minute timestep and the CICE took a 30-minute timestep. There were no restoring functions used. These plots show the Arctic ice thickness in meters for October of year 10, and the Antarctic ice thickness for December. These simulations were performed on the IBM p690 (cheetah) at ORNL.



8.0 LAND SURFACE MODEL AND RIVER TRANSPORT MODEL (TOPIC COORDINATOR: F. HOFFMAN)

Background

This chapter describes software engineering and development tasks for the land surface model component of the CCSM. CLM is the land model for CCSM and the CAM. A collaborative project between the Terrestrial Sciences Section of the Climate and Global Dynamics Division (CGD) at NCAR and the CCSM Land Model Working Group (LMWG), CLM formalizes and quantifies concepts of ecological climatology. It simulates the physical, chemical, and biological processes by which terrestrial ecosystems affect and are affected by climate across a variety of spatial and temporal scales. CLM consists of components for biogeophysics, hydrology and river transport, biogeochemistry, and dynamic vegetation.

The latest version of CLM, called CLM2.1, incorporates a hierarchical subgrid data structure composed of grid cells, land units, columns, and plant functional types (PFTs). Each grid cell contains one or more land units, each land unit contains one or more columns, and each column contains one or more PFTs. The land unit is intended to capture the broadest spatial patterns of physically distinct surface types including bare ground, vegetation, lakes, and glaciers. The column subdivides the potential variability in the soil and snow state variables within a single land unit while the PFT further subdivides columns based on their biophysical and biogeochemical differences due to broad categories of vegetation.

Objectives

The objectives of the land surface efforts are to establish and improve software design and documentation, develop and implement selected frameworks and interfaces for utilities and communications, optimize the performance of the model on target platforms and new computer architectures, improve subgrid scale parameterizations, and provide support to NCAR and the CCSM LMWG. Identified as high priority tasks early in the project were: the development of a Requirements Document, the design and implementation of a flexible cache-friendly decomposition, and the improvement of the subgrid scale orography for precipitation and runoff In addition, the design, development, and implementation of a new coupling interface between the land model and the atmosphere model was suggested as a high priority task by NCAR. This coupling interface is used when CAM is run in standalone mode since it requires the land model in order to run. In the first half of the project, effort was focused on these tasks.

Identified more recently as an objective is the vectorization of the land model for use on the ES and the Cray X1 architectures. Desired is a single source code which will give reasonable performance on both scalar and vector machines. Because the data structures and code were developed for scalar platforms, vectorization of the model requires implementation of new data structures and modifications to all parts of the existing model code. Significant effort has been applied to this task, and completion of an initial version of the scalar/vector-friendly CLM code is targeted for mid-October 2003.

Progress

The CCSM SEWG examines software-related issues in CCSM and recommends software engineering practices to support CCSM project goals. SEWG develops and maintains the CCSM Software Engineering Plan, the CCSM Software Developer's Guide, and the CCSM Software Coordination Plan. The Software Developer's Guide includes recommendations for documents

designed to enumerate the goals and functionality of software development and direct software engineering activities. These documents include a Requirements Document to establish required functionality; an Architecture Document to describe the overall structure of a software component; a Detailed Design Document used by developers to plan and develop internal and external interfaces, a User's Guide to explain how to build and run the software, and a User's Reference Manual to describe the functions, interfaces, and use of the software. The Requirements Document describes the scientific and computational requirements of a CCSM component model, and links these requirements to project requirements as appropriate. Primarily the responsibility of the LMWG scientists, it is used to help direct future software development activities for the land model.

A Requirements Document was drafted and presented to the LMWG on March 28, 2002. After undergoing an extended review, the final document was accepted by the LMWG at the CCSM Annual Meeting in June 2002. The document is available in both HTML and PostScript formats on the CLM web page at NCAR.

8.1 CACHE-FRIENDLY DECOMPOSITION AND LOAD BALANCING

In a pre-cursor to this project, a number of performance optimizations were made to the atmosphere model. The physics in CAM was modified to group atmosphere columns into "chunks." The size of these chunks is chosen to optimize CPU cache utilization. Since chunks are distributed among available MPI processes, they serve as the basis for parallel decomposition. A similar strategy for decomposition has been adopted and implemented for CLM.

Grid cells in CLM, the highest level in the subgrid scale hierarchy (see the left half of Figure 22), correspond to physics columns in the atmosphere. A new data structure called "clumps" was created to group land grid cells into cache-friendly portions and to provide a load-balanced decomposition of the land surface. Clumps contain a varying number of grid cells because the work associated with each is proportional to the number of PFTs contained therein. Because a grid cell can not span multiple clumps, a perfect load balance may not be achieved depending on the distribution of PFTs among grid cells.

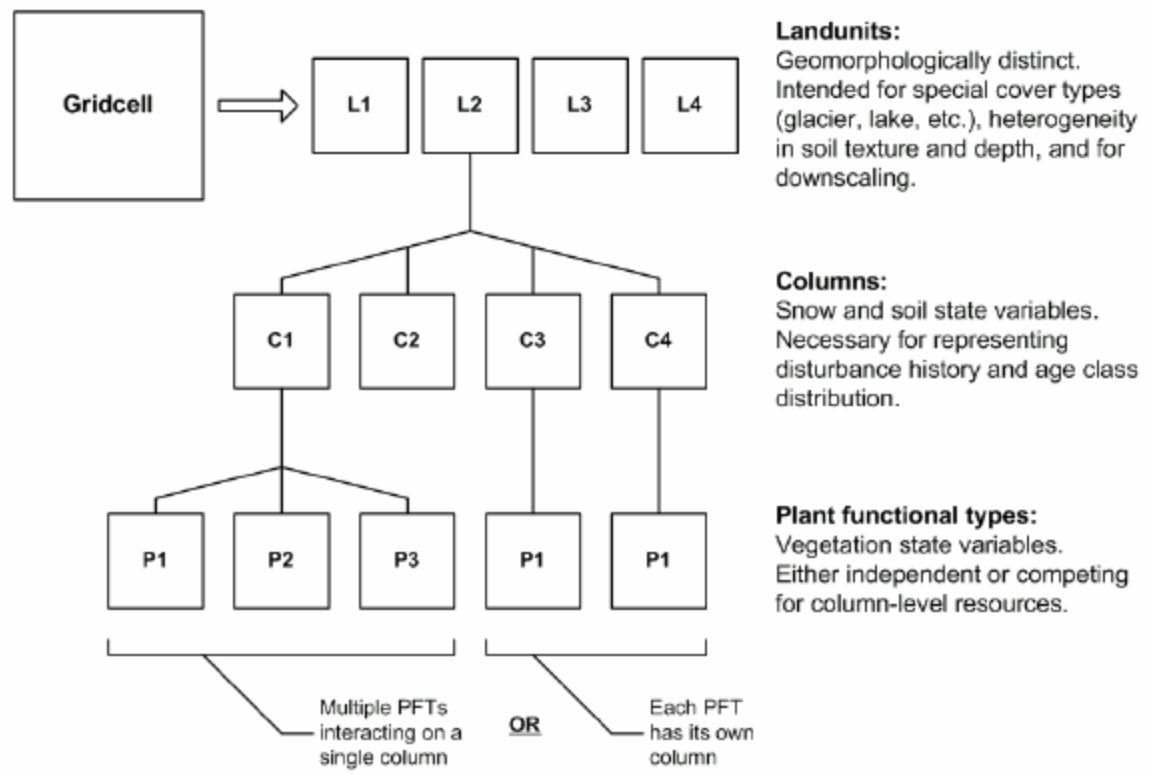


Fig. 22. Subgrid scale hierarchy used in CLM 2.1

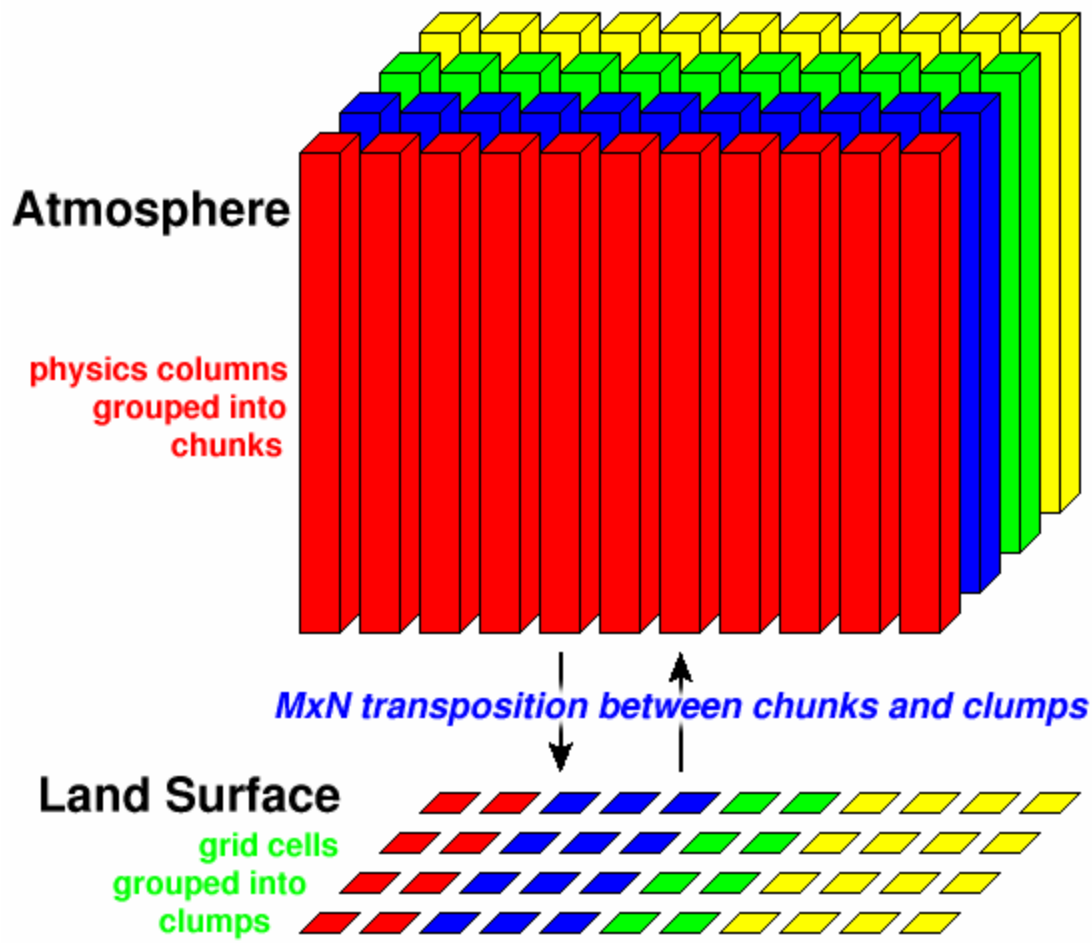


Fig. 23. Mapping of physics columns grouped into chunks to land surface grid cells grouped into clumps.

8.2 NEW ATMOSPHERE/LAND COUPLING INTERFACE

CLM may be run in offline mode, coupled directly with CAM, or coupled with CCSM component models by way of the flux coupler. When CAM is run in offline studies, CLM is usually run with it to provide land/atmosphere interactions. When run in this mode on parallel computers, the models run as a single executable and share the same processors. Surface fluxes and states are exchanged between the land and the atmosphere using MPI. In CAM2.0.1 these data exchanges occurred by way of a single MPI process; i.e., all surface data (including data over the oceans) were sent from the atmosphere physics to a single process which subsequently redistributed the land data to the appropriate MPI processes. This bottleneck could consume a significant amount of run time, particularly as models resolutions and processor counts increase.

A method of exchanging only the land surface fluxes and states directly between the appropriate CAM and CLM MPI processes was developed and implemented. A new module called *lp_coupling* manages the process mapping between atmosphere chunks and land clumps (see the right half of Figure 23). Two new data structures, *clump2chunk* and *chunk2clump*, are initialized after both the atmosphere and the land models have performed their decompositions.

These data structures may be re-initialized at any time if the physics or land decompositions change thereby supporting future dynamic load balancing algorithms. Multiple MPI gathers and scatters were replaced by a single *MPI_alltoallv()* before and after each call to the land model *driver()* routine. Fluxes and states for atmosphere columns over the ocean are no longer exchanged with the land model.

Performance tests for the new interface were run on the IBM Power 4 at Oak Ridge using a modified CAM2.0.1 with CLM2.0 at T42 resolution and 26 levels with the Eulerian dynamical core for 720 time steps. Mean total communications time was reduced by a factor of 7 to 9, depending on configuration. Since communication time is attributed to the land model, the improved interface reduces total land model run time. The total land model runtime consumed by communications was significantly reduced (see Figure 24). Even greater performance improvements will be realized at higher resolutions. On 256 processors of the IBM Power 4 at ORNL with a CAM chunk size of 16 (pcols=16), the new CAM/CLM interface increases the number of simulation years per day from 31 to 38 (see Figure 25). The new interface was made available to the public in the CAM2.0.2 release.

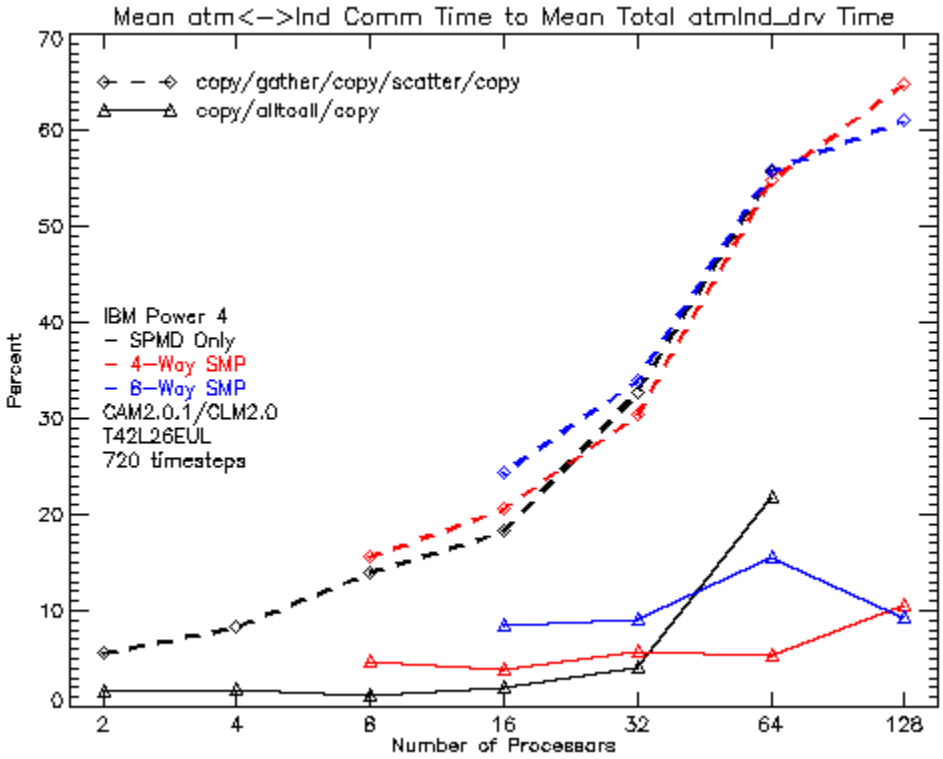


Fig. 24. Mean atmosphere/land communications time to mean total land model run time.

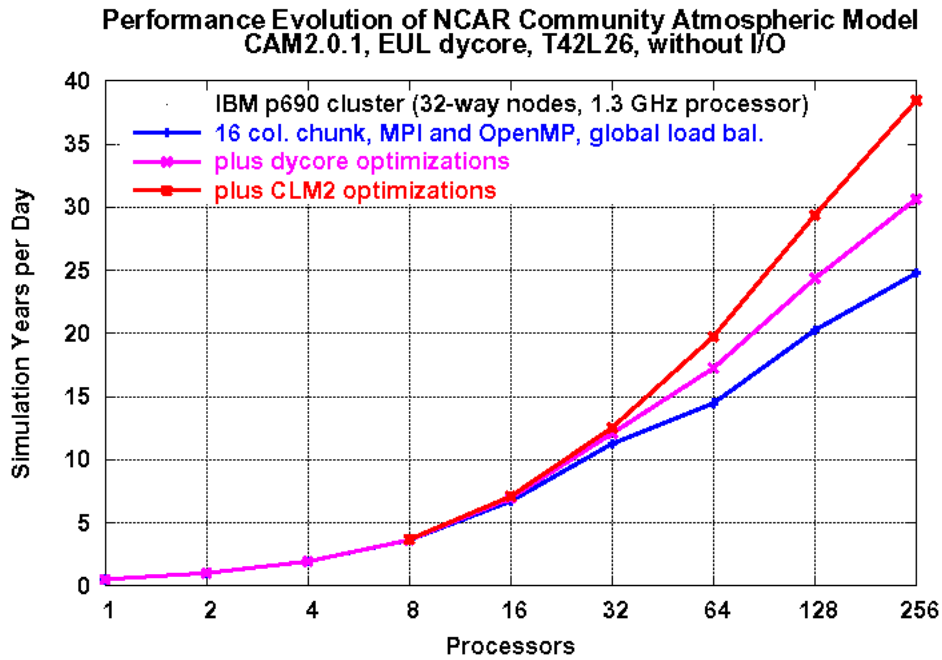


Fig. 25. The new interface significantly reduces the percentage of total land model run time consumed by communications. The new CAM/CLM interface increases the number of simulations years per day for CAM from 31 to 38.

8.3 SUB-GRID SCALE OROGRAPHY AND RUNOFF

To maintain consistency with the treatment of the atmosphere, the Ghan et al. (2002) treatment of the subgrid influence of topography on the atmosphere has also been applied to the land surface. Thus, the atmospheric conditions (temperature, moisture, precipitation, radiative fluxes) predicted in the lowest layer for each elevation class are supplied to the land model for each elevation class. For simplicity the surface properties (plant functional types and soil types) are assumed to be the same for each elevation class, but the elevation classification is applied to each land model subgrid surface type within each land model grid cell. The different atmospheric conditions predicted for each elevation class produce a different land surface behavior for each elevation class, with colder temperatures, and more precipitation, snow water, and runoff in the higher elevation classes. This offers the potential to significantly improve the simulation of surface runoff and river discharge by the CCSM. We have therefore added an option to the River Transport Model (RTM) (which uses a grid that is independent of the atmosphere and land) in which the surface runoff is distributed according to the mean elevation of each RTM grid cell. Care is taken to ensure that runoff is conserved by the mapping procedure. A new five minute river direction dataset has been developed to make better use of the improved runoff prediction. To facilitate diagnosis of the RTM performance, a monthly RTM history option has also been added.

8.4 ADAPTATION TO NEW HARDWARE

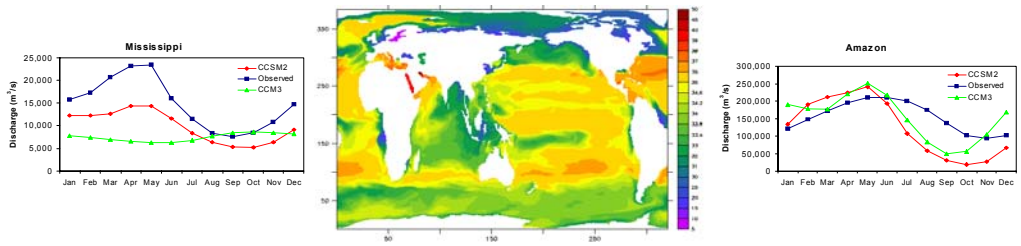
The Earth Simulator (NEC SX-6) in Japan has sparked renewed interest in vector computers among researchers at NCAR as well as others in the CCSM community. DOE's new Cray X1 at ORNL is a second vector platform expected to perform climate simulations. This growing

interest in vector machines has created a need for vectorizable model code. Unfortunately, CLM2.1 was designed with massively parallel scalar machines in mind so it is vector-hostile. We have investigated a variety of options for making CLM vector-friendly while preserving (or improving) performance on existing target platforms.

In CLM 2.1, the internal data structures are based on a hierarchy of pointers to derived data types containing scalar quantities scattered throughout memory. New data structures were required if the model were to utilize vector pipelines. A number of options were presented to NCAR scientists, and a choice was made in June 2003. The LMWG adopted the proposed data structure changes, and a new branch was created in the CVS repository for vector development. The structural changes required to the code are significant. Loops over columns must be moved into the science subroutines. Nearly every line of code must be "touched" in order ensure vectorization. This development is nearly complete, and is targeted for completion in early October 2003. This schedule will beat the end-of-calendar-year deadline after which the attention of much of the community will be focused on IPCC runs.

8.5 ANALYSIS OF RUNOFF IN CCSM2 CONTROL SIMULATION

A 350- year control simulation of CCSM2 was analyzed with regard to continental runoff effects. Results were compared to observational data and previous simulations using CCM3. River discharge into the Arctic Ocean improved significantly over earlier versions. The Niger, St. Lawrence and Congo rivers were among those that did not show appropriate discharge into the oceans. Sea surface salinity responded with decereased salinity near river outlet points. The July sea surface salinity is shown for year 320 of the control simulation.



Plans

For the remainder of the project, work will continue in software design and documentation, framework implementation, performance optimization, and subgrid parameterization. A Software Design Document will be created based on the new hybrid scalar/vector code to document development strategies and lessons learned in the vectorization process. Additional effort will focus on improving code modularity and utilization of frameworks like ESMF. Moreover, improvements in load balancing may be required when dynamic vegetation is fully-enabled in CLM, and performance optimization will likely be needed as new biogeochemistry routines are added at the end of the year. Adaptation to new hardware will always be ongoing as computer systems evolve. Finally, additional work in river transport modeling will be performed to improve the performance and science results from the energy and constituent exchange between component models attributed to surface runoff.

9.0 ATMOSPHERIC CHEMISTRY (TOPIC COORDINATOR: P. CAMERON-SMITH)

Background

Since pre-industrial times, the concentrations of various aerosol types (e.g., sulfate, black carbon, and mineral dust) and several key greenhouse gases such as methane (CH_4), nitrous oxide (N_2O), and ozone (O_3), have been changing because of anthropogenic activities. Collectively, the magnitude of the climate forcing from these species is larger than that of carbon dioxide (CO_2) although some are positive and some are negative (see Figure 26).

The behavior and effect of these non-CO_2 species is more complicated than for CO_2 because they are affected by atmospheric chemistry and aerosol microphysics, so their distributions are more heterogeneous. There are also feedbacks between climate, chemistry, and aerosols that further increase the importance of chemistry and aerosols, *e.g.* a change in any one of stratospheric ozone, stratospheric temperature, or stratospheric dynamics will feedback on the other two. For aerosols, in addition to the direct effect of scattering and absorbing light, they act indirectly by serving as cloud condensation nuclei (CCN), leading to clouds with more (but smaller) droplets that reflect more sunlight and last longer, thus cooling the atmosphere. Aerosols and atmospheric chemistry can also have an impact through interaction with the biosphere, *e.g.,* fertilization of the land with nitrogen species and fertilization of the oceans with Iron from mineral dust. There is also chemical production of CO_2 in the atmosphere through oxidation of species such as CH_4, CO and turpenes. Thus, to predict and understand future climates, the radiative forcing from these non-CO_2 gases and aerosols, as well as their feedbacks into the radiative, dynamical, and biogeochemical balances, must be taken into account.

The non-CO_2 species are also important because they should be more amenable to anthropogenic control measures trying to mitigate climate change (Hansen *et al*., 2000) than CO_2 because they have shorter atmospheric lifetimes.

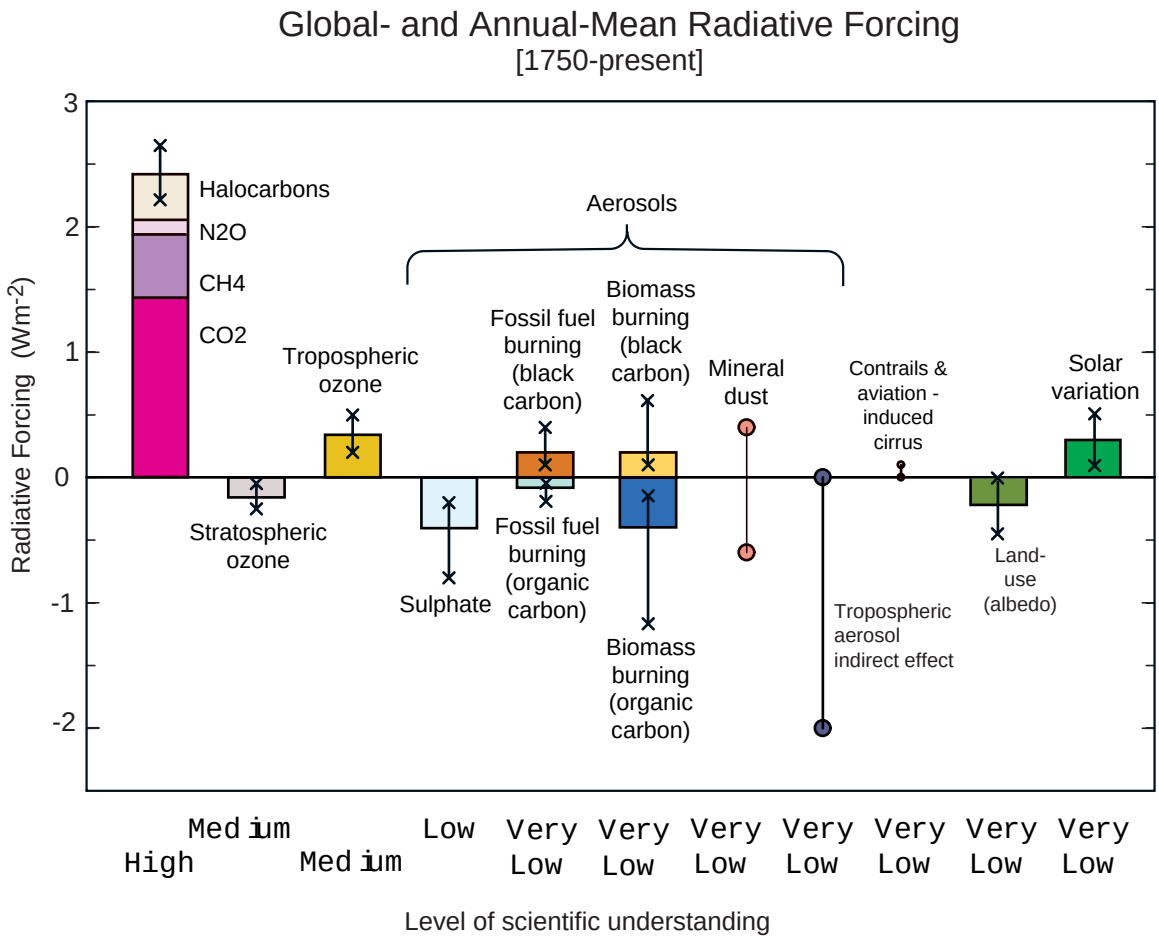


Fig. 26. Global annual mean radiative forcing (Wm^{-2}) from IPCC (2001).

Objectives

Broadly, the goal of this work is to implement interactive ozone photochemistry, methane chemistry, and aerosols into the CCSM in a way that is computationally efficient yet accurate for climate modeling needs. The original objectives were to implement and validate tropospheric-only chemistry in the CCSM model by the end of 2003, followed by implementation and validation of a combined stratosphere-troposphere chemistry capability by the start of 2005, and addition of aerosol microphysics by the end of the project in 2006. This work would leverage the ongoing WACCM effort at NCAR (a combined CGD and ACD effort) and build upon the off-line chemistry models at LLNL (IMPACT model) and NCAR (MOZART model).

The goals also include the implementation in CCSM of MAGPI (a marine aerosol and gas phase interactions code developed by David Erickson at ORNL), and thereby link atmospheric chemistry and sulfate aerosols to ocean biogeochemistry.

The main change from the original plan was to leapfrog the development of a troposphere-only model and directly implement two chemical mechanisms that could simultaneously simulate the stratosphere and troposphere. The first mechanism is small and fast for use in general CCSM simulations, while the second mechanism is more extensive for studying chemistry-climate interactions. This has meant we are now ahead of our schedule for implementing the combined

stratosphere-troposphere mechanism in 2005, and we have two mechanisms instead of one, thereby expanding the utility of chemistry within the CCSM model. There is still considerable work to be done testing and validating the atmospheric chemistry code, particularly with respect to its influence on climate in long 21st century simulations where the feedbacks will be strongest.

The objectives for implementing aerosols in the original proposal were fairly general since this work was in the out years and the modeling of aerosols has been advancing. In discussions with Bill Collins (Chair of SSC committee) and Phil Rasch (NCAR) we have identified five long-term aerosol goals for CCSM. The exact role we will play under SciDAC, and how we can leverage our existing expertise with aerosols in the IMPACT model, is still under discussion. The five aerosol goals are:

1. Adding a bulk formulation for aerosol types that have not yet been implemented completely in CCSM (nitrates, sea-salt, and secondary organics).
2. Introducing size resolved treatments, including microphysics, for all the major aerosols.
3. Treating the mixing state of the aerosols (important for their radiative effects and evolution).
4. Introducing the 1st and 2nd indirect effects on liquid clouds.
5. Parameterizing the longwave effects of soil dust and possibly sea salt.

We also plan to go beyond the usual method of validating and evaluating aerosol and chemistry models (i.e., comparison of species concentrations with observations) by analyzing the response of the atmospheric model (CAM2) to simulations with and without our additional models using the standard experimental protocol of the Atmospheric Model Intercomparison Project (AMIP simulations are constrained by observed monthly mean sea-surface temperatures and sea-ice, starting in 1979 and run to near-present). Extensive diagnostic tests will be performed on these runs using diagnostic tools developed by the DOE PCMDI, and will be done in collaboration with Peter Gleckler at PCMDI.

This work is well aligned with the objectives of NCAR and CCSM, and has been guided by discussions with Bill Collins (Chair of the SSC committee). The incorporation of both atmospheric chemistry and interactive aerosols into CCSM are major goals of recent CCSM science plans (CCSM science plan 2004-2008, CCSM plan 2000-2005). Indeed, the latest plan (CCSM science plan 2004-2008) states:

- "Because the chemical processes in the troposphere and stratosphere are highly non-linear, it is crucial to have a realistic characterization of the chemical species and their reactions."
- "Current models ... do not include feedbacks between ozone and dynamics. This deficiency needs to be corrected."
- "Given the multifaceted roles of aerosols in the climate system, it is imperative that the capability to model a fully interactive aerosol system be developed within CCSM."

Progress

9.1 CCSM DEVELOPMENT (LAMARQUE, WALTERS, KINNISON, MCKENNA)

A gas-phase chemistry package suitable for tropospheric and stratospheric conditions has been implemented in the WACCM version of CAM2. Simulations were performed and analysis of the results is underway. The chemical scheme offers a complete description of hydrocarbon oxidation in the troposphere and of stratospheric ozone chemistry in the stratosphere. This

package includes emissions, deposition (wet and dry), transport (large and subgrid-scale), and photochemical reactions for 106 species. Photolytic reaction rates are calculated using an approach similar to the LLNL Look-Up-Table. Wet and dry deposition algorithms were taken from the MOZART model. All algorithms have been implemented into the WACCM version CAM2. Because our scientific interest is in the troposphere and stratosphere (not higher), we have reduced the model extent to approximately 85 km. This altitude provides a natural boundary across which limited chemical transport occurs. The chemically influenced fields (such as ozone, CFCs, etc.) are fed back to the climate model and used in the radiative calculations.

In the process of integrating interactive chemistry into the climate model, we have performed a set of simulations to evaluate against measurements the performance of the model with interactive chemistry. These simulations were performed at the resolution of 2°x2.5^{o} with 52 levels and for 15 years. No drift was found in the simulated ozone field, indicating the lack of misrepresented chemical mechanisms. The analysis of the simulated ozone field (Fig. 28) for a variety of stations indicate that the model is performing very similarly to a tropospheric CTM (MOZART-2) that has the same set of chemical species and reactions. The shortcomings of the simulations are therefore related to problems in emissions and/or chemistry, not in the coupling between the chemistry and the climate models.

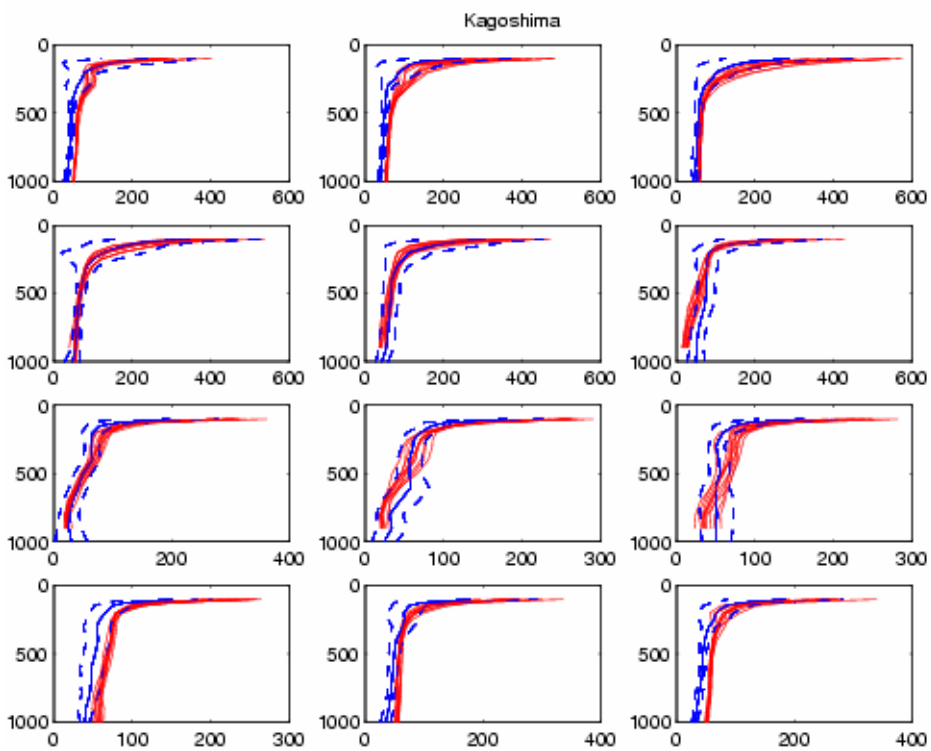


Fig. 27. Comparison of modeled ozone (red lines, one per simulation year) with ozonesondes data at Kagoshima (Japan) (blue line, dash line is one standard deviation. Each box is per month, January to December (top left to bottom right). Vertical axis is pressure (hPa) and horizontal axis is mixing ratio (ppbv).

In addition to ozone, we have performed the comparison of the model results for other chemical species against surface (CO from CMDL) aircraft measurements during specific campaigns. These results (Figures 28, 29, and 30) indicate that the model performs reasonably well in the

simulation of more reactive species such as NO_x. Evaluation of the stratospheric portion of the model is underway.

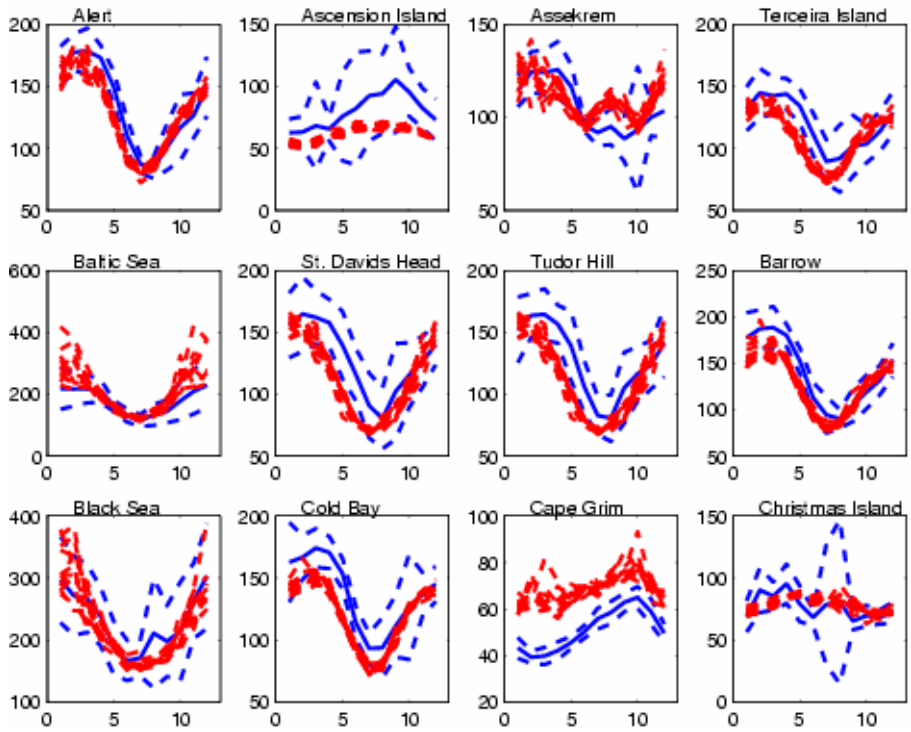


Fig. 28. Comparison of modeled CO (red lines, one per simulation year) with surface CMDL with a variety of stations. Horizontal axis is month of the year and vertical axis is mixing ratio (ppbv).

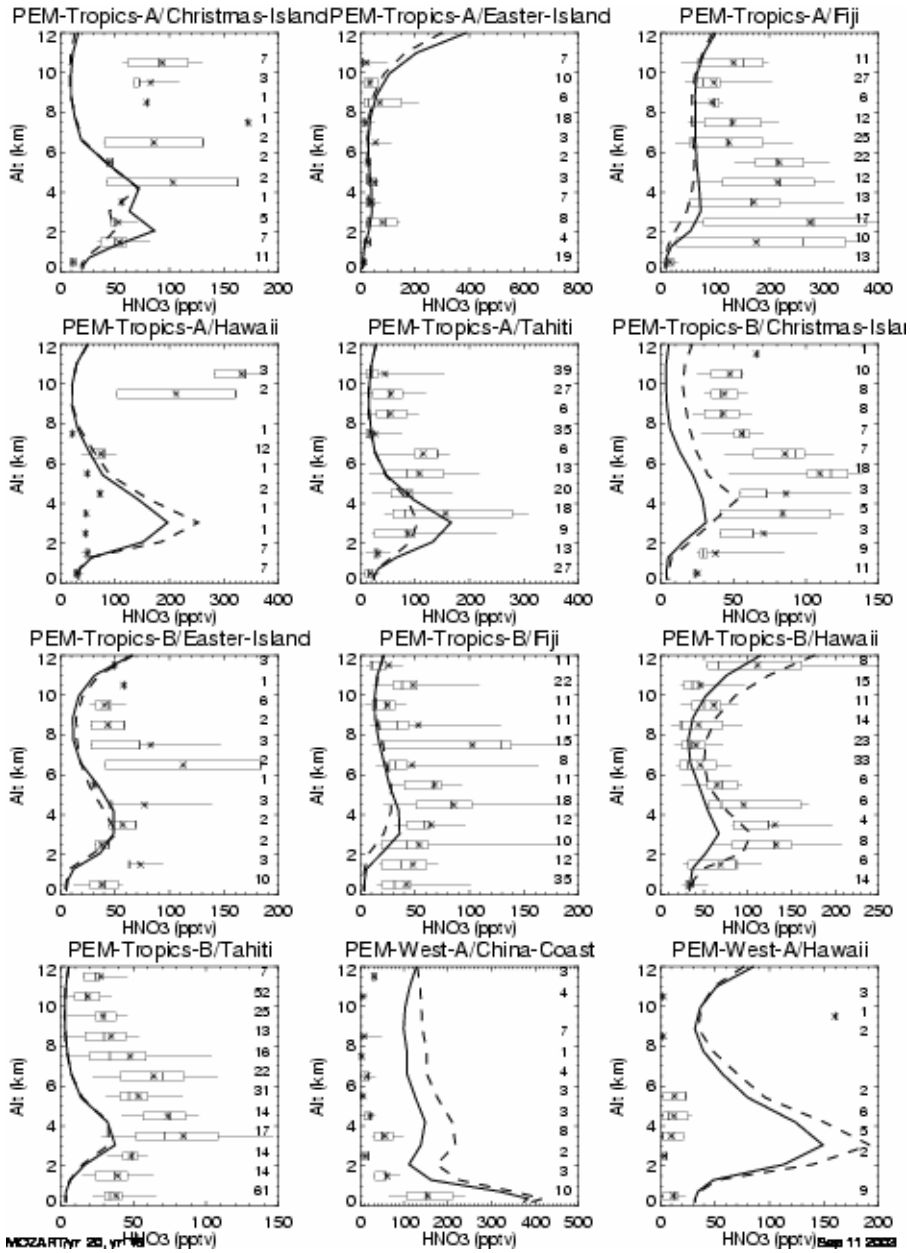


Fig. 29. Comparison of HNO3 mixing ratio with selected aircraft campaigns. Solid (dashed) line is model results for year 15 (14). The number of measurements is indicated on the right-hand side of each plot.

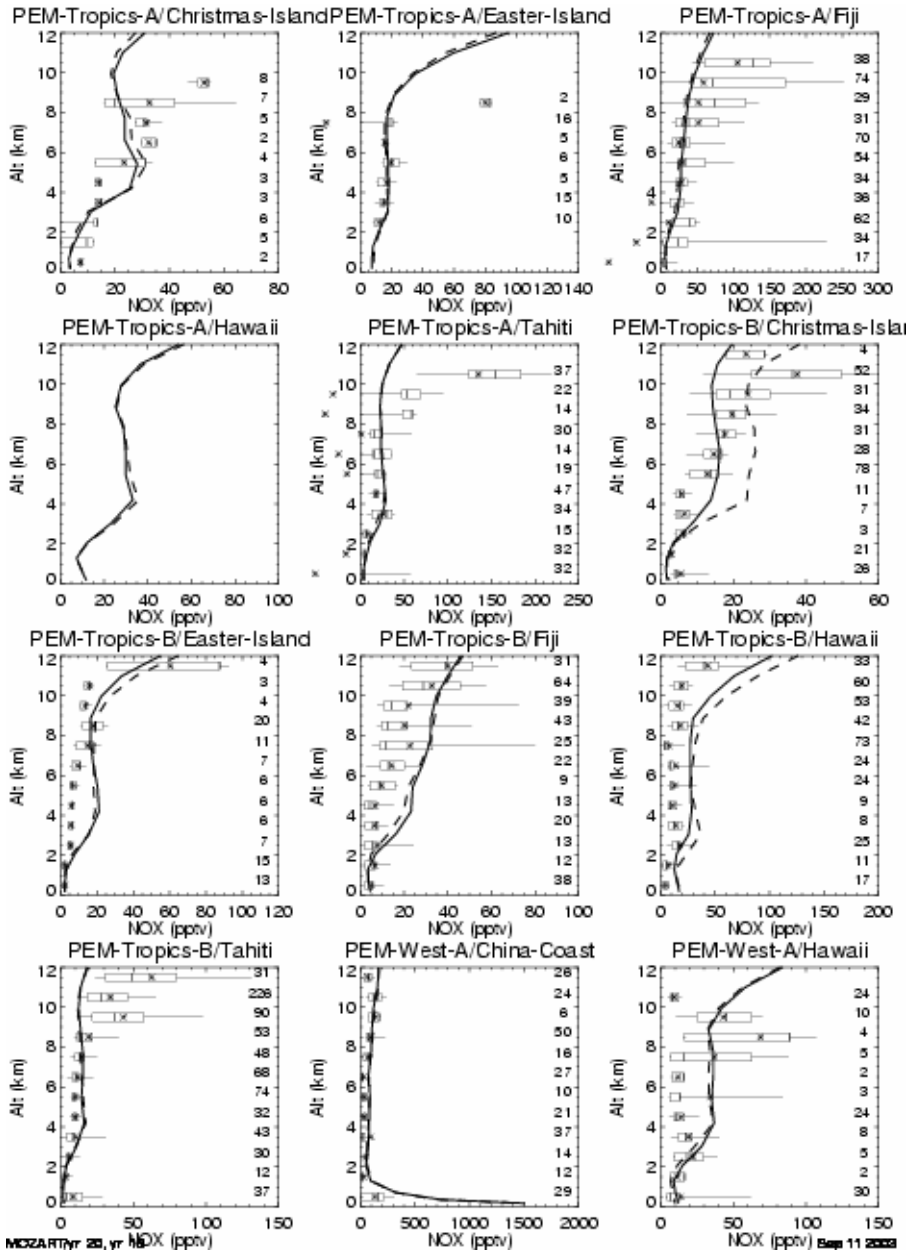


Fig. 30. Comparison of NO_x mixing ratio with selected aircraft campaigns. Solid (dashed) line is model results for year 15 (14). The number of measurements is indicated on the right-hand side of each plot.

Although this model is quite expensive to run (it takes approximately one wall-clock day per simulation year on a IBM-SP Power-4 96 CPUs, roughly doubling the cost of the equivalent WACCM model without chemistry), it provides a very realistic description of the average chemical state of the troposphere. This will be the reference to which the reduced mechanism (see below) will be compared.

We are now in the process of having interactive emissions of biogenic VOCs from the CLM used as boundary conditions for this model. This new feature will provide another coupling between chemistry and climate.

9.2 CHEMICAL MECHANISM DEVELOPMENT (CAMERON-SMITH, CONNELL, ROTMAN, TANNAHILL)

To enable efficient chemistry simulations within multi-century climate simulations a fast compact chemistry mechanism that simulates both the troposphere and stratosphere has been developed that should provide CCSM with reasonable heating rates due to atmospheric chemistry. Since it is physically based it should also respond reasonably under altered climate conditions.

Since the last report in December 2002 we have improved the fidelity of the compact mechanism (previously the compact mechanism was only within a factor of 2 in the troposphere and now it agrees within 20%), and ported the compact mechanism to the CCSM where it is undergoing testing. We have also implemented long and short wave radiation packages into the IMPACT model so we can validate the heating rates being generated by the chemistry within CCSM once the compact mechanism is running.

The compact mechanism was developed by paring down our state-of-the-art chemistry mechanism (named TS2), which includes Ox, HOy, NOy, ClOy, BrOy, SOx, and VOC chemistry for both the troposphere and stratosphere, in a way that realistically calculates ozone around the tropopause, where it produces the greatest radiative forcing. The compact mechanism (named TS4) includes essential HOx and NOy reactions, is four times smaller and faster than our full mechanism (TS2), and covers both the troposphere and stratosphere.

The magnitude of the overhead ozone is very important when calculating photolysis rates. With the full mechanism (TS2) we use the ozone field calculated by the IMPACT model (Rotman, *et al*., 2003), so the model is fully interactive in this respect. However, with the small mechanism (TS4) we were concerned that a less accurate ozone field might feedback onto photolysis rates. Hence, we tested the small mechanism using both its own ozone field and ozone climatology for photolysis (the photolysis package used in IMPACT is the Look-Up-Table as used above in the CAM runs).

To test the compact mechanism we ran three 10-year simulations in the IMPACT model:

1. Compact mechanism (TS4), using the ozone field it generates for photolysis rates.
2. Compact mechanism (TS4), using the ozone climatology for photolysis rates.
3. Full mechanism (TS2), using the ozone field it generates for photolysis rates.

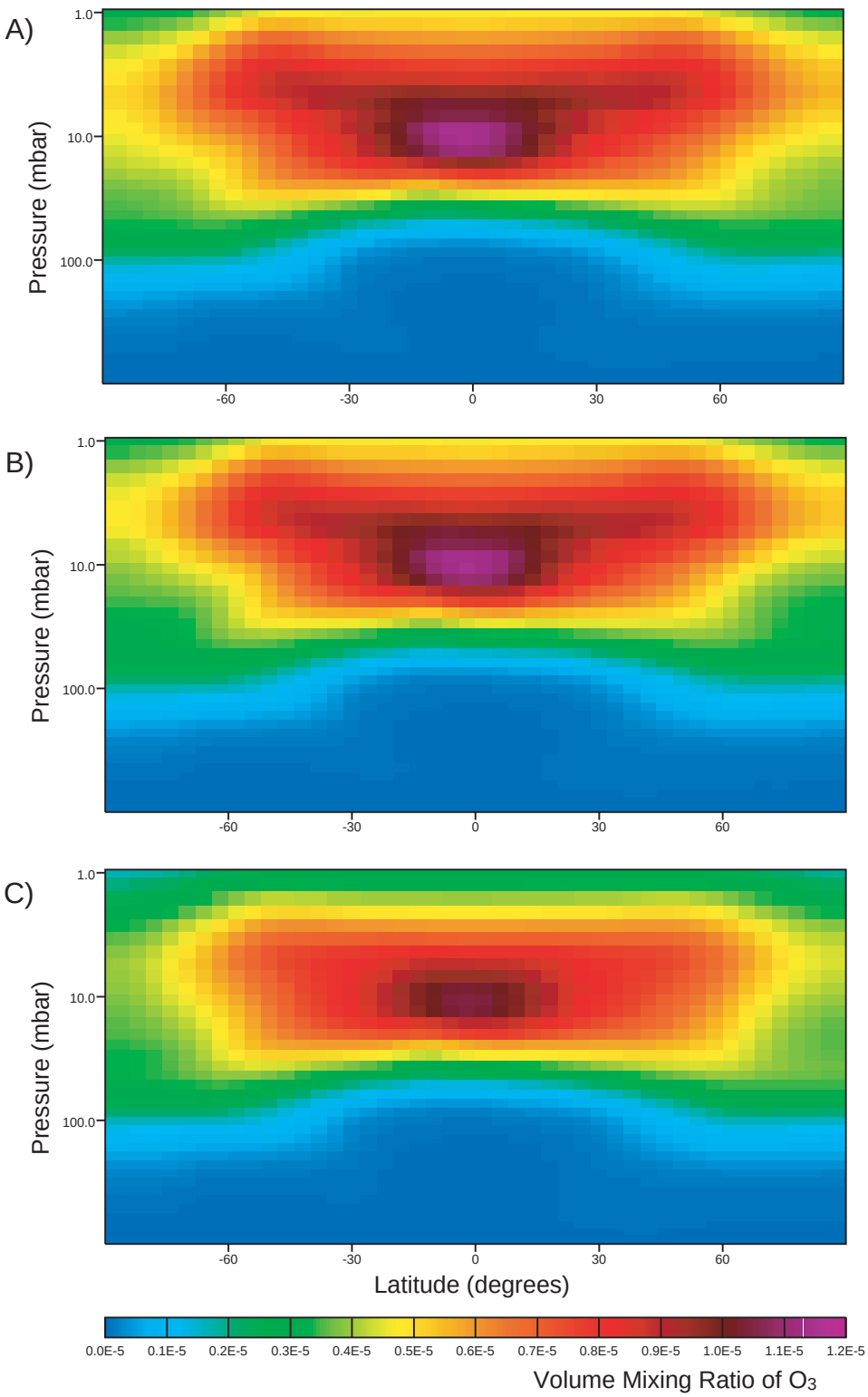


Fig. 31. Volume mixing ratio of ozone in the tropopause region for simulations A, B, and C.

The morphology of the ozone distribution for each of the small mechanism runs (Figure 31) is reasonable throughout the troposphere and stratosphere. We quantified the comparison using the ratio of zonal mean ozone for each of the small mechanisms to the full mechanism (Figure 32). The concentration of ozone is now within a factor of 20% throughout the troposphere and stratosphere (as opposed to a factor of 2 previously), and is even closer in the critical tropopause region.

Overall, the compact mechanism using the ozone climatology for photolysis gives a slightly better fit to reality, but the compact mechanism using its own ozone still does a good job, and has the advantage that it will respond to altered climates more consistently.

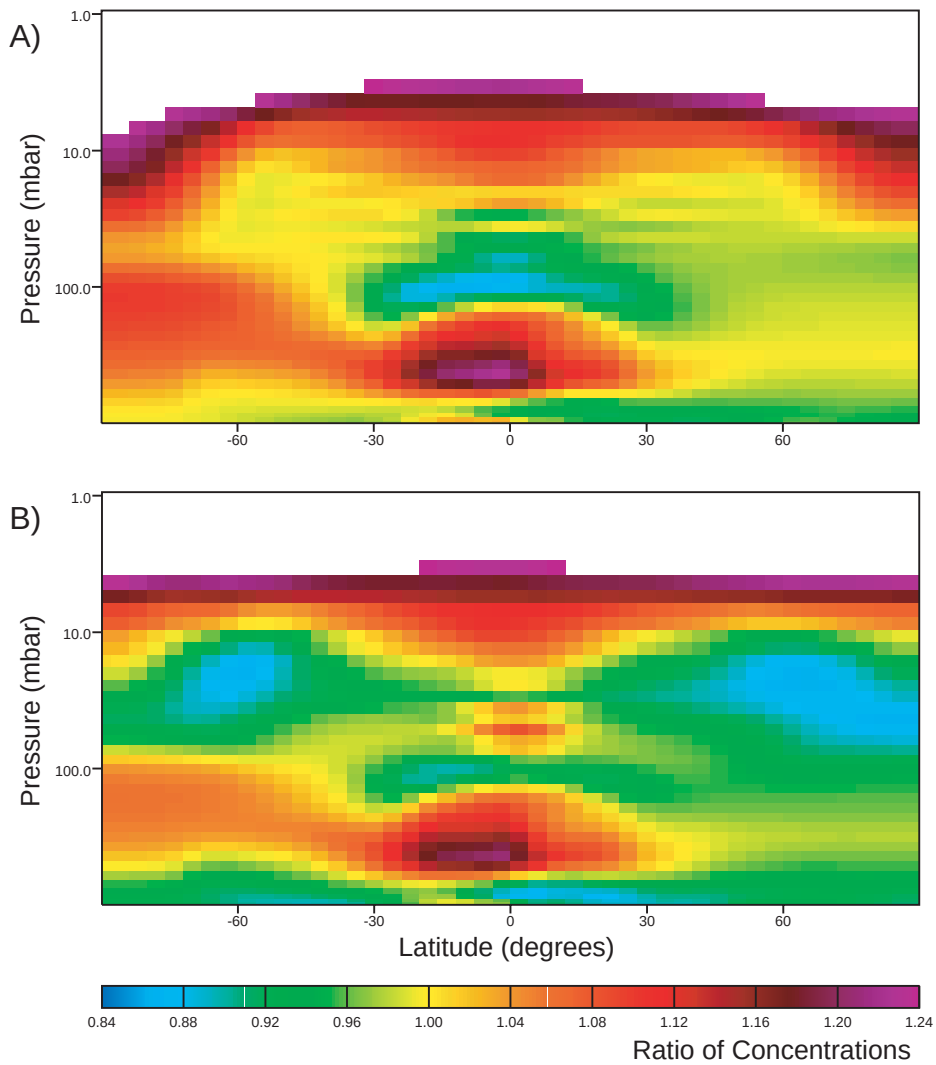


Fig. 32. Ratio of zonal mean ozone in July for small chemistry funs to full chemistry run (i.e., A/C and B/C). Note that the scale on these plots is considerably better than in our previous report, where the compact mechanism only agreed with the full mechanism within a factor of 2 in the troposphere and stratosphere.

9.3 SOURCES AND SINKS OF ATMOSPHERIC METHANE (TAYLOR, CAMERON-SMITH)

Atmospheric methane is the second most important greenhouse gas contributing to anthropogenic climate change. Increasing atmospheric methane concentrations affect the chemistry of the troposphere and stratosphere. As a first step towards the eventual development of interactive methane fluxes, we are currently developing a comprehensive set of source and sink fluxes suitable for inclusion into CCSM based on past research (Taylor et al., 1991). We are currently working on combining previously developed estimates of the sources and sinks of atmospheric methane with more recently derived estimates of atmospheric methane included as part of publicly available data bases. Code will eventually need to be developed to read this data into CCSM and incorporate the fluxes into the atmospheric chemistry code. We will be performing continuing analyses of the distributions and fluxes of methane and carbon monoxide, and their role in influencing the distribution of ozone in the troposphere and stratosphere, as the chemistry solver and chemical mechanism are developed and incorporated into CCSM.

We have completed a comprehensive set of baseline IMPACT model output data for analysis. We have recently completed a new set of baseline runs with modified initial methane concentrations reflecting the observed methane concentration for the mid 1990's. Based on these model runs we have been performing a series of runs aimed at calibrating the methane fluxes in order to produce a growth rate in methane concentration comparable to that observed during the 1990's. These simulations are performed with a state-of-the-art combined troposphere-stratosphere chemical mechanism (TS2, Rotman, et al., 2003). This analysis provides a baseline case for future analyses as we develop the compact mechanism (TS4) and ensure that we do not induce significant model artifacts associated with the compact chemical mechanism. We have nearly completed performing the analysis of the off-line model simulations using the LLNL IMPACT model and are planning to publish the results of our work to date. We are currently porting a version of the WACCM model, as provided by NCAR, to the IBM computers at ORNL and the Jazz Linux cluster at ANL to which we will incorporate our methane fluxes. The results of our work to date are available on the web site located at http://www-climate.mcs.anl.gov/proj/climate/public_html/climate-SciDAC.html

The results under the [CH4] tag examine a comparison between the original methane emissions included in the IMPACT model, the new methane emissions data developed as part of the SciDAC project, and Climate Monitoring & Diagnostics Laboratory (CMDL) observations. The new emission fields produce a substantial improvement in comparison with CH4 measurements made at CMDL monitoring sites, particularly in the Northern Hemisphere. Figure 33 illustrates the improvement in predicted methane concentrations observed at Barrow, Alaska. We have also investigated the effect of changed methane emissions on OH concentrations (see the [OH] tag at the web site above). Figure 34 illustrates a comparison of OH concentrations, again at Barrow Alaska. These results reveal that the changes in methane concentration shown in Figure 33 are primarily due to changes in methane emissions rather than feedbacks with OH concentration.

Figure 35 presents a comparison of the seasonal cycle in carbon monoxide (CO) concentrations derived from the IMPACT model obtained using the two methane emissions estimates along with CO observations obtained by CMDL. Changes in methane emissions can affect CO concentrations by changing the rate of production of CO from the oxidation of methane and via changes in OH that alter the rate of oxidation of CO. Figure 35 shows that carbon monoxide concentrations produced by IMPACT are nearly identical for the two methane emissions fields. This result is consistent with the small difference observed in hydroxyl radical concentrations generated by the IMPACT model, as shown in Figure 34.

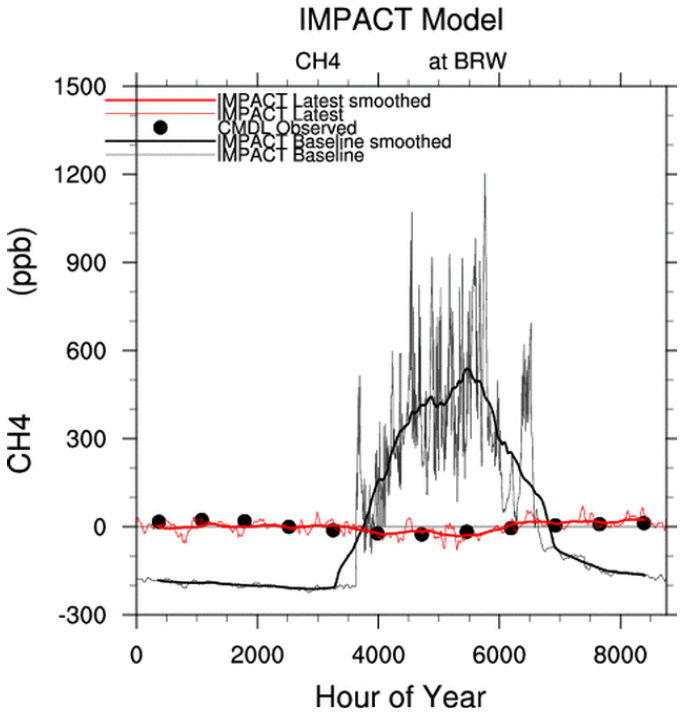


Fig. 33. Comparison of deviations (from mean) of methane concentrations derived from the IMPACT model obtained using the original methane emissions (baseline) and the methane emissions developed as part of the SciDAC project (latest). The latest methane emissions included in the IMPACT model produce methane concentrations that more closely match seasonal variations in measured concentrations and show much lower variability.

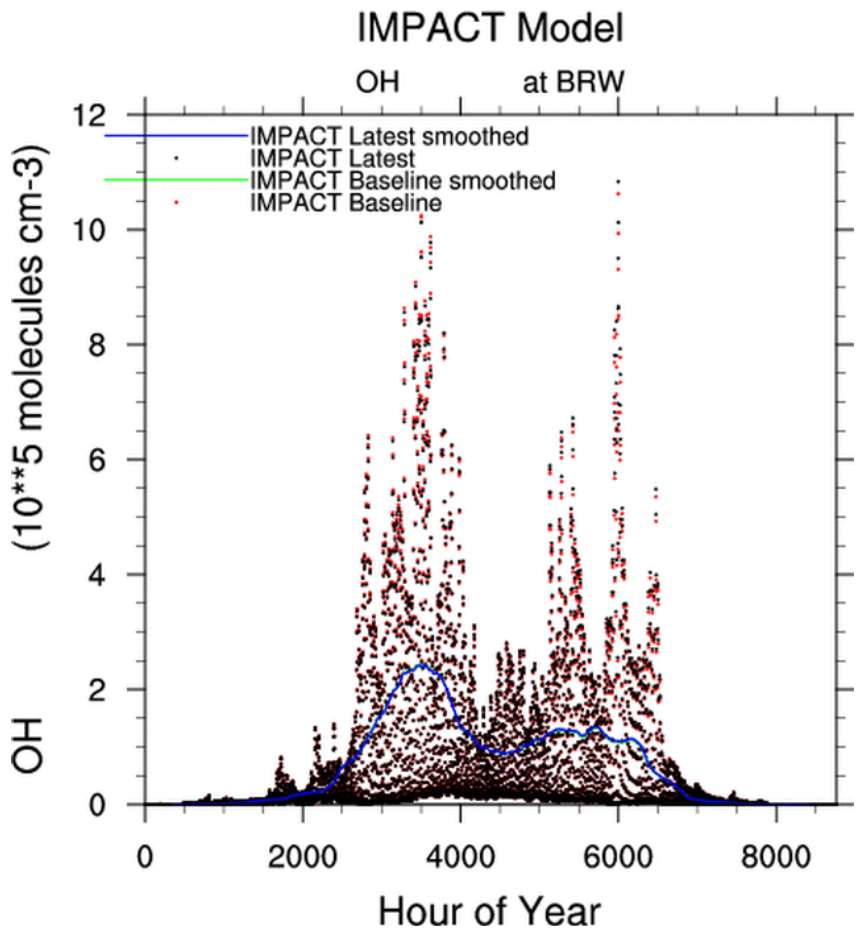


Fig. 34. Comparison of OH concentrations derived from the IMPACT model obtained using the original methane emissions (baseline) and the methane emissions developed as part of the SciDAC project (latest). OH concentrations show only small changes, mostly at high OH values, so we can conclude that the changes in methane concentration shown in Figure 2 are primarily due to changes in methane emissions rather than feedbacks with OH concentration.

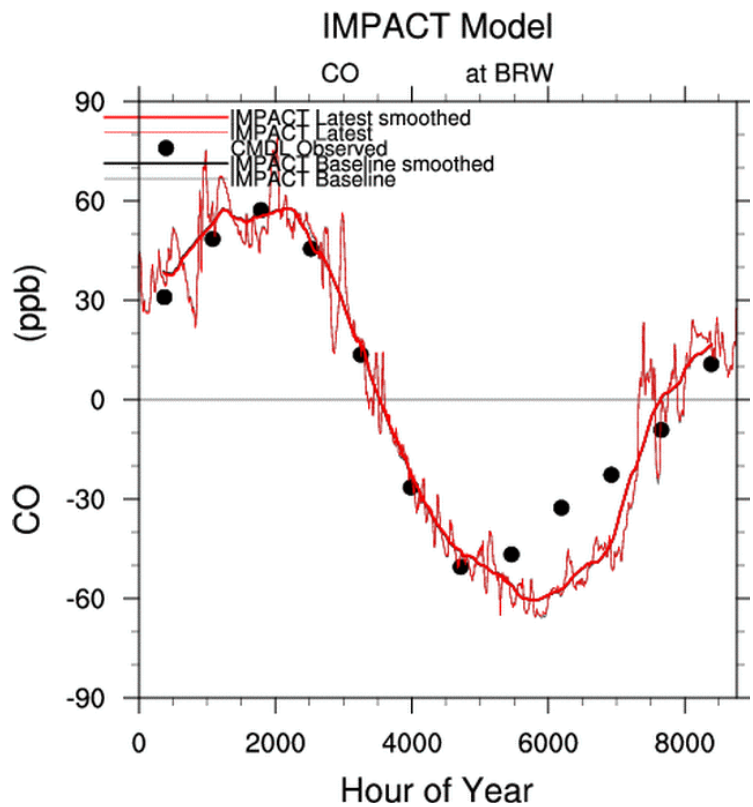


Fig. 35. Comparison of carbon monoxide concentrations derived from the IMPACT model obtained using the original methane emissions (baseline) in comparison with the carbon monoxide concentrations obtained using the new methane emissions. Carbon monoxide concentrations are nearly identical.

9.4 MARINE AEROSOL AND GAS PHASE INTERACTIONS (ERICKSON, TAYLOR)

On the marine aerosol and gas phase interactions MAGPI code (Erickson et al., 1999), an initial code redesign included making the code FORTRAN 90 compliant, breaking the code into subroutines representing the functionality of the code, adding Protex headers and reformatting the code consistent with the NCAR documentation requirements. We have recently completed a major modification to this code by adding the capability to use a GEAR ODE solver, consistent with the solvers included in WACCM. The GEAR method is better suited for long simulations with CCSM. This new code base is a needed first step in the transition of the MAGPI model into CCSM and is nearing completion. The next step will be to integrate this new MAGPI code into WACCM.

We have also completed a suite of initial test runs with the new code and we are analyzing the output of these runs. We have also developed a new analysis code for this purpose. The results of our work to date are located at http://www-climate.mcs.anl.gov/proj/climate/public_html/climate-SciDAC.html under the MAGPI tag.

Results are consistent with the original MAGPI code. Figure 36 shows the output of the MAGPI model for the nitrogen species, both in the gas phase and in the aqueous phase on the aerosol particle as a function of time.

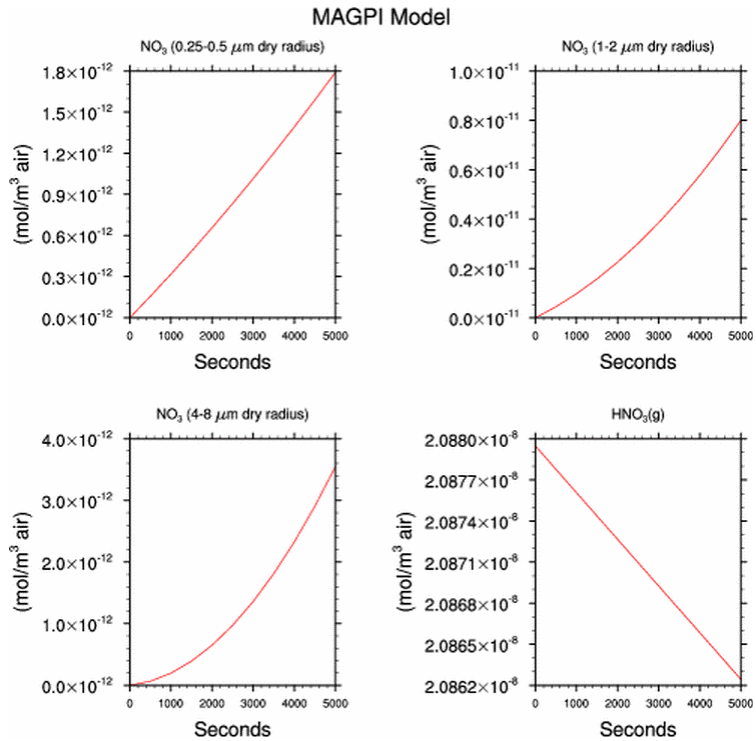


Fig. 36. Variation in the concentration of nitrogen species in the MAGPI model in both the gas and aqueous phases as a function of time.

9.5 MODEL VALIDATION (CAMERON-SMITH, LAMARQUE, TAYLOR)

We are validating the model results by comparing predicted concentrations with *in situ* observations and the original models that have already been validated (IMPACT, MOZART, and MAGPI). Because the direct coupling from chemistry to the atmospheric model is through radiation, we also need to check that the small mechanism produces heating rates that are comparable to the full mechanism. To this end we have implemented short-wave and long-wave radiation packages into IMPACT. Preliminary results for the heating rate produced by ozone are shown in Figure 37.

We also plan to analyze the response of the atmospheric model (CAM2) to simulations with and without our additional models using the standard experimental protocol of the AMIP- (AMIP simulations are constrained by observed monthly mean sea-surface temperatures and sea-ice, starting in 1979 and run to near-present). Extensive diagnostic tests will be performed on these runs using diagnostic tools developed by the DOE PCMDI, and will be done in collaboration with Peter Gleckler at PCMDI.

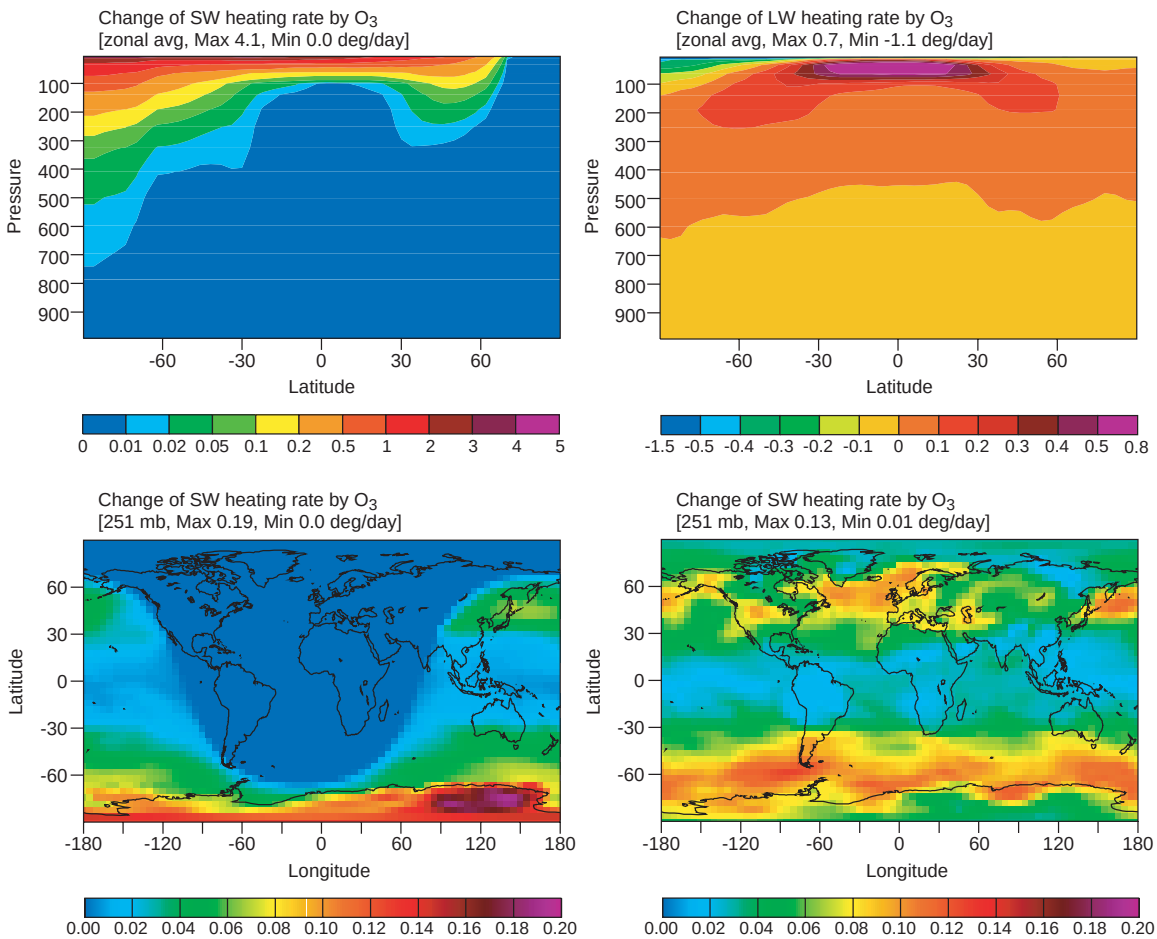


Fig. 37. Preliminary plot of heating rate due to ozone in the short-wave (0.175 - 4 microns) and long-wave (longward of 4 microns) for midnight GMT, January 1. The upper plots are zonal mean heating rates. The lower plots are on the 251 mbar surface.

Is this rate of progress adequate to meet the objectives within the 5 year period?

Leap-frogging the tropospheric-only chemical model means that we are ahead of our target for our atmospheric chemistry objectives, although there is plenty of work left to do validating the atmospheric chemistry.

The work involved in the CCSM aerosol goals listed under Objectives (above) is far beyond the scope of this SciDAC project. We are therefore in discussion with Bill Collins (Chair of SSC committee) to identify the most useful role we can play under SciDAC for the duration of this project.

What objectives have been completed?

We have completed the addition of the MOZART chemical code and combined stratosphere-troposphere mechanism to CCSM. A run of 20 years has been completed and evaluation is in progress. We have also developed a fast compact mechanism in the IMPACT model and ported it to the new CCSM chemistry mechanism format. This code is currently being tested in CCSM.

The new methane emission database has been constructed, and is currently being refined in the IMPACT model to ensure it reproduces observed trends.

The MAGPI code has been made compliant with CCSM standards and a GEAR ODE solver has been incorporated.

Plans

In line with our original plan, our priorities for atmospheric chemistry are the completion, inclusion into CCSM, and validation of the two atmospheric chemistry mechanisms, the methane emission data set, and the MAGPI model.

Also in line with our original plan, we will increase our efforts on implementing aerosols into the CCSM. The exact role we will play under SciDAC, in terms of how we will collaborate with NCAR and which tasks we will complete by the end of this project, is still under discussion with NCAR and the CCSM project. The topics under consideration include: addition of incompletely implemented aerosol species, aerosol microphysics, the 1st and 2nd indirect effects on liquid clouds, and parameterizing the longwave effects of soil dust and sea salt (see Objectives section above for details).

We will validate each contribution to CCSM through comparison of model species concentrations with *in situ* and satellite observations, and compare the response of the atmospheric model (CAM2) by performing AMIP-type simulations and extensive diagnostic tests developed by PCMDI (see Objectives section above for details).

We will collaborate with the following groups:

1. NCAR WACCM: implementing atmospheric chemistry.
2. NCAR (ACD and CGD: Collins, Rasch, Tie): implementing atmospheric aerosols into CCSM.
3. DOE PCMDI (Peter Gleckler): testing and validating the effect of aerosols and chemistry on the atmospheric model through AMIP style simulations.
4. Ocean biogeochemistry portion of this SciDAC project: to improve the computational structure of MAGPI (a marine aerosol and gas-phase interactions code developed by David Erickson at ORNL); to use DMS generated by the ocean biogeochemical model in the atmospheric chemistry and aerosol code; and to provide aeolian dust output to the ocean biogeochemistry model for iron fertilization.

10.0 BIOGEOCHEMISTRY (TOPIC COORDINATOR: D. ERICKSON)

Background

The role for feedbacks in the biogeochemical system that impact climate prediction has been recognized as being critical to climate modeling. As outlined in the original proposal, the identification and implementation of biogeochemical cycles and feedbacks in the CCSM2 model is imperative to future climate predictions.

It should be noted that the biogeochemistry task significantly overlaps the atmospheric chemistry task and also is directly related to the land and ocean tasks. Essentially, the land and ocean both contain extensive biogeochemical systems that inject trace gas molecules, including CO2 and aerosols into the atmosphere. These trace gases and aerosols influence atmospheric radiation and by implication computed climate statistics.

Objectives

The main objectives of the Biogeochemistry topic is to extend the CCSM2 physical model to include germane ecological and biogeochemical feedbacks in the climate system. This requires that oceanic ecosystem models be included in the physical ocean model. Changes in the oceanic physical circulation has an impact on the ability of the ocean to take up CO2. This coupling of the global climate and carbon systems is a critical aspect of future climate prediction over the next 10-50 years.

Progress

The first objective for this task was to implement an ocean biogeochemistry module in the POP model. Nutrients and sulfur has been added to the POP model. A carbon cycle has been added to those of the nutrients and sulfur. The diatoms have been segregated as a separate tracer and plant taxon, such that a silicon cycle is included as well. DMS distributions are improved at the larger scales by specifying species-dependent, cell-internal dimethyl sulfonium propionate concentrations and by introducing first-order bacterial consumption. Improved geocycling and ecodynamics are now run on the fine grid. PCO_2 and dimethyl sulfide results are compared with the available climatologies. The SCIDAC biogeochemistry team has in coarse resolution OGCM frameworks simulated the surface dynamics of other trace gases which cross the sea-air interface to enter the lower atmospheric photochemical system. Carbonyl sulfide, the methyl halides, and

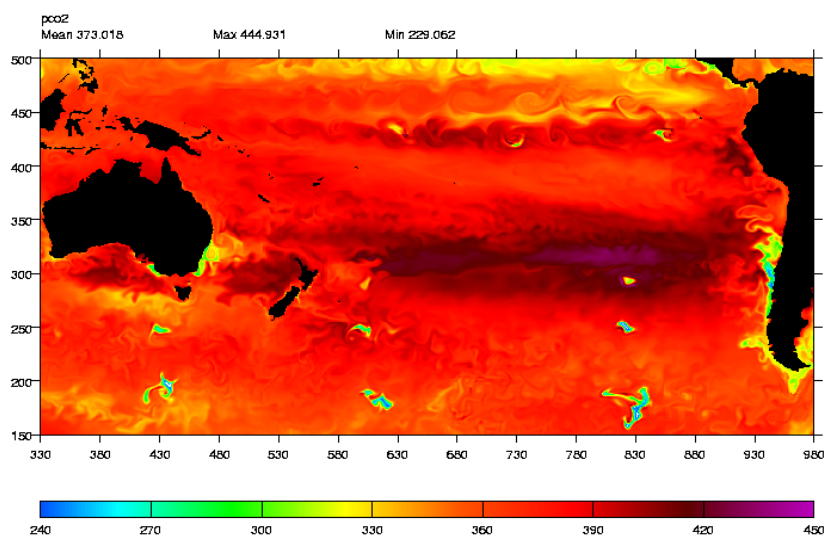


Fig. 38. Surface layer carbon dioxide partial pressure field simulated for the date January 31, 1999 in the biogeochemical Parallel Ocean Program. Background strongly resembles the standard Takahashi data set but with mesoscale eddies resolved. Multiple patch iron enrichments introduced into Southern Hemispheric High Nutrient Low Chlorophyll Zones on January 1 have filamented and also caused significant carbon drawdown.

simple nonmethane hydrocarbons have all been represented successfully as they are produced by phytoplankton or the photolysis of dissolved organics. Precursor and product species are included as well.

10.1 FINE-RESOLUTION POP SIMULATIONS (S. ELLIOT, S. CHU, M. MALTRUD)

Prior to obtaining SciDAC support, the LANL ocean biogeochemistry team had succeeded under internal funding in simulating global surface chlorophyll distributions based on coupled nitrogen/iron nutrient ecodynamics. During early SciDAC sponsored work carbon, silicon and sulfur cycles were appended to the biogeochemical version POP. Most recently the new additions have been analyzed and refined in several key aspects. Carbon dioxide partial pressure values have been compared with the standard Takahashi data set. Improvements in the geochemical initial and boundary conditions have brought about agreement at the percent level. The carbon and silicon cycles have required enhanced taxonomic resolution in the phytoplankton bins. Diatoms and coccolithophorids are now treated independently of smaller species. It has been learned that more sophisticated zooplankton grazing parameterizations will also be required for consistency. Sulfur cycling includes intracellular (phytoplanktonic) DMSP and exudation/processing of DMS. Bacterial uptake has been modified to reflect variation in strength of the microbial loop across ecological provinces. POP computed DMS distributions have been converted to sea-air fluxes by collaborators at ORNL and agree well with the available climatologies both locally and on an integrated basis. Under SciDAC external support, multiple patch iron enrichment scenarios are being studied and demonstrate the capability for biogeochemical POP applications in environmental engineering. SciDAC relevant mechanism adjustments have been identified and will be incorporated into next generation runs. They involve the incorporation of further detail in the trace gas transfer schemes and a reduction in iron scavenging (inorganic removal) time constants. To date Los Alamos has relied on UC Berkeley Aeolian iron deposition data sets. ORNL collaborators have expertise in this area so that there will soon be a transition to DOE generated dust input climatologies. Other nonCO2 trace gases tracked in the ecodynamical POP include methyl bromide and the nonmethane hydrocarbons. SciDAC supports global tropospheric photochemistry simulations as well. Discussion is underway with LLNL collaborators regarding coupling of surface ocean trace gas processing to lower atmospheric chemistry models. All this occurs by way of preparation for more general SciDAC tasks, which encompass the addition of bacterial and trace gas processing to the CCSM framework.

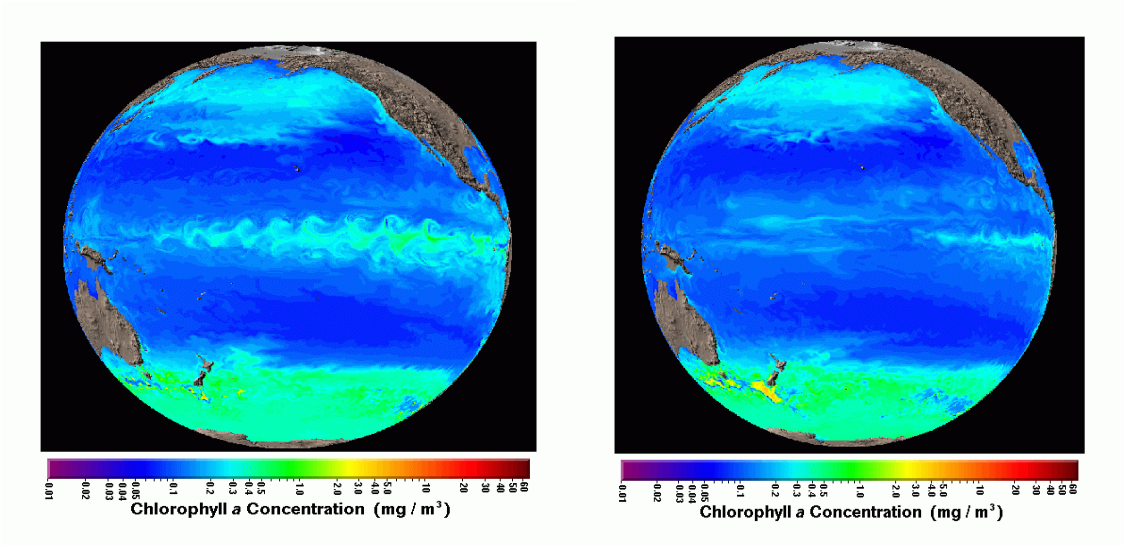


Fig. 39. Two snapshots show surface chlorophyll distributions simulated with the biogeochemical version of the POP for conditions in late 1996 (a La Niña year) and late 1997 (full El Niño conditions). Biological activity is intense across the equatorial Pacific during La Niña (Dec. 96), extending almost from the primary Peruvian upwelling zone to New Guinea. The warm pool then shifts eastward, shutting off the upwelling of nutrient-rich cold water. Plankton growth slows and the chlorophyll peak gradually recedes towards the east (Dec. 97). The standard SeaWiFS remote-sensing color bar is used. Comparisons with satellite ocean color measurements have validated model accuracy. Units are milligrams chlorophyll per cubic meter in all cases.

10.2 IRON ENRICHMENT STUDIES (M. MULTRUD)

Iron enrichment may be viewed in the real ocean and in numerical models as ecodynamical perturbation experiments or as preparation for carbon management. In either case, LANL global ocean simulations will permit the most complete statistical analyses. Here we display the evolution in the biogeochemical version of POP of three patch-type iron-enrichments introduced south of New Zealand, roughly bracketing the locations of recent SOFEX expeditions (Southern Ocean Iron experiments). Iron is introduced in the model as roughly one-nanomolar iron injections over three areas, each approximately 100 km on a side. The first image shows the resulting distribution of iron after one month of processing by surface ocean biology transport. The second image is the corresponding phytoplankton distribution (units of micromolar plant nitrogen content). Complex biogeochemical waves are apparent, probably the result of differential zooplankton grazing effects.

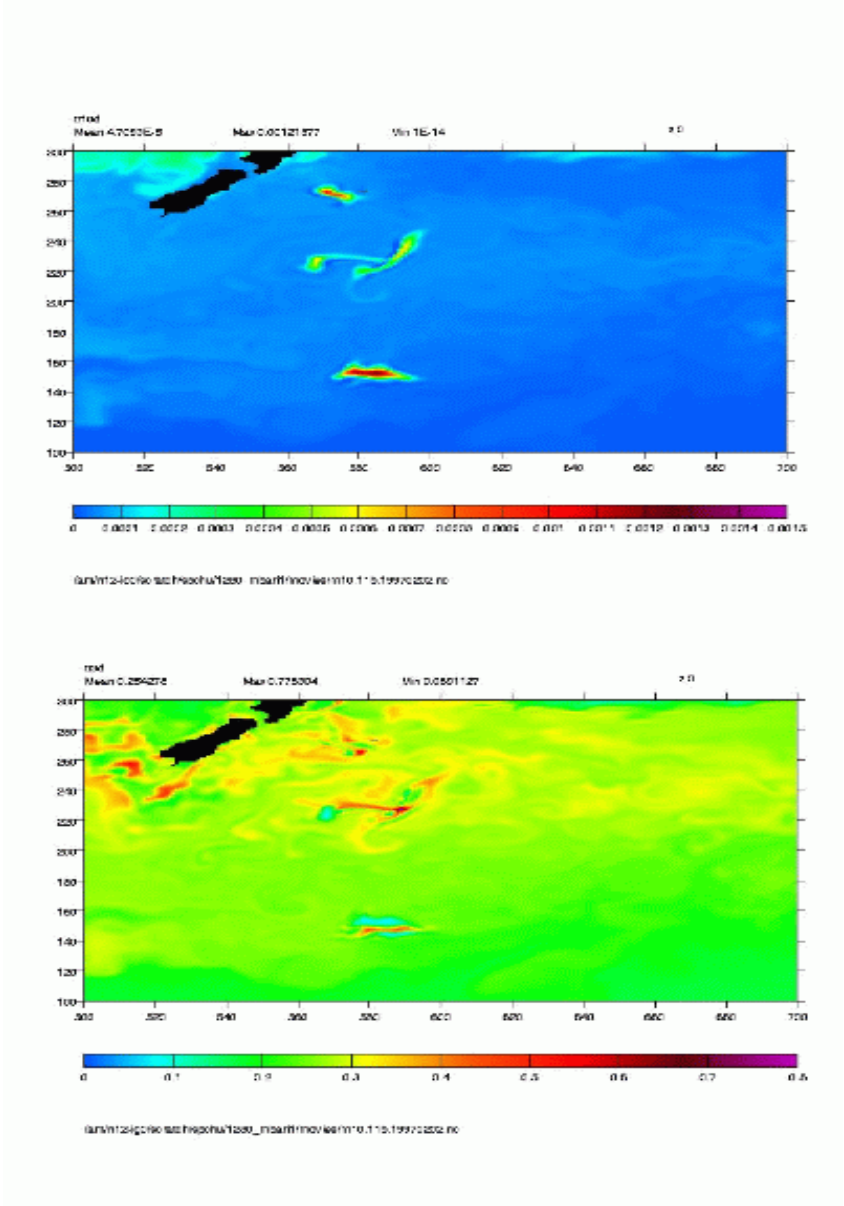
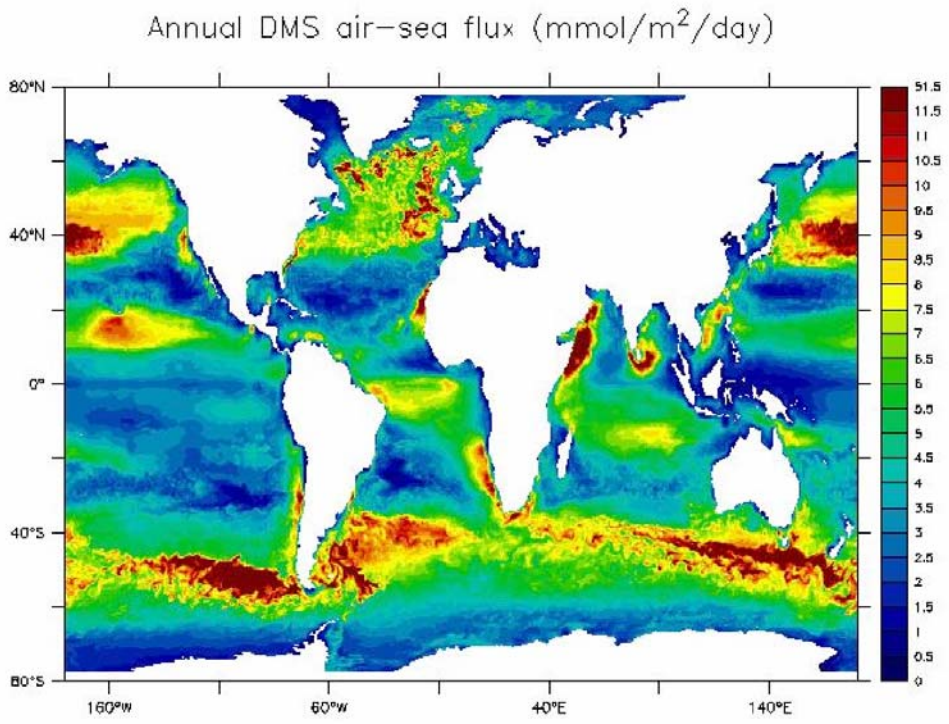


Fig. 40. Iron is introduced in the model as roughly one-nanomolar iron injections over three areas, each approximately 100 km on a side. The first image shows the resulting distribution of iron after one month of processing by surface ocean biology transport. The second image is the corresponding phytoplankton distribution (units of micromolar plant nitrogen content).

10.3 GLOBAL DMS FLUX FROM THE OCEAN USING POP (SHAOPING CHU, MAT MULTRUD, SCOTT ELLIOTT, JOSE HERNANDEZ, DAVID ERICKSON)

DMS, a precursor of atmospheric sulfate aerosols over some 70% of the Earth, has been coupled with detailed models of air-sea gas flux and are now available to the atmospheric chemistry community. The Fig. below shows the annual ocean to air flux of DMS as computed via the ecosystem model in POP.



We are currently embarking upon research, which will be required to add surface ocean cycling for the molecules carbon monoxide, isoprene, methane and nitrous oxide. The plan is to test all the model trace gases in a coarse mode POP mixed layer, then move the routines into the CCSM framework. These first few runs will be conducted with concentration boundary conditions prescribed throughout the lowest layer of the atmosphere. During the next 2 years of SciDAC activity, the plan is to relax the boundary constraints by coupling with simplified atmospheric photochemistry.

Eddy driven variability in pCO2, DMS, CH3Br and CH3I fluxes from the ocean to the atmosphere is important. The need to put the oceanic fluxes of trace gases and particles into atmospheric chemistry/climate models was noted. The linkage between the oceanic carbon cycle and other trace gases is strong and links between climate change, carbon cycle dynamics and fisheries and ocean management needs to be assessed.

Atmospheric chemistry needs to include ocean, land fluxes within the climate model addressing tropospheric ozone, N2O etc. Tropospheric ozone impacts vegetation and the carbon cycle and has an impact on carbon sequestration. As an example, China nitrogen, ozone and Fe fluxes will evolve over next 100 years and need to be assessed. Fe is critical to oceanic carbon budget and climate induced changes in Fe fluxes change carbon and climate. Several SCIDAC scientists are

involved in atmospheric chemistry model construction and validation and the biogeochemical model formulations will influence atmospheric chemistry.

Research plans

The ability to swap in and out i.e., convection, plug and play different ocean models, atmospheric models, and dynamical cores, ecosystems on land and ocean, and biogeochemistry modules is required. A high level of automation and different chemical solvers need to be evaluated. High-resolution simulations are a requirement for biogeochemical cycles just as for the purely physical models and are as high a priority in biogeochemical simulations as any other CCSM2 component. There is a need to increase resolution due to the intrinsic heterogeneous of biology and chemistry. A detailed biogeochemical grid required. Climate modeling must be done at global climate model at 10 kilometer resolution. Better comparison to station observations is required.

The SciDAC project scientists are actively involved with the CCSM Biogeochemistry working group. 4-5 scientists attend the CCSM Biogeochemistry working group meetings. It was recognized that SciDAC needs to contribute on a fundamental level to the CCSM2 model development and that is being done via the carbon cycle, trace gas fluxes, atmospheric chemistry and evaluation of feedbacks in coupled climate-carbon models. Acclimation, diffuse radiation and other basic climate science issues are being addressed.

The basic research plan is to refine the ocean biogeochemical parameterizations under development and extend the compounds to include halogenated compounds and aerosol fluxes. These ocean sources of trace species will be included in the source terms of the atmospheric chemistry module. Also, as the continental regions experience alterations in temperature and soil moisture, this will change the atmospheric source of Fe to the ocean thus impacting CO2 uptake. An important component of the research plan is to clearly identify the feedbacks in the climate – biogeochemical system and include these in future climate prediction simulations.

10.4 BIOGEOCHEMISTRY DIAGNOSTICS (K. CALDEIRA, S. DONEY)

Background

Ocean biogeochemistry is important if we are to model climate/carbon-cycle feedbacks and better understand the consequences of climate and chemical changes on the marine environment. NCAR is leading an effort to develop a marine biogeochemistry model for use in CCSM. It is necessary to evaluate this model in the light of observations, both to understand the reliability of this model and to help point to ways in which the model could be improved.

Objectives

Initially, we had proposed to help NCAR develop aspects of their biogeochemistry model, specifically, the representation of deep-ocean processes and aspects of some elemental cycles, as well as to help with diagnostics for the ocean biogeochemistry component of CCSM.

In conversation with Scott Doney, at that time leader of NCAR's ocean biogeochemistry modeling effort, it was decided that LLNL's effort in this area could most usefully play a role in the SciDAC-CCSM collaboration by developing diagnostics for the evaluation of marine biogeochemical simulations and apply those diagnostics to the biogeochemical results of CCSM.

Our project will contribute greatly to meeting NCAR and CCSM objectives. The primary goal of this effort is to develop a system wherein NCAR (and other institutions) can quantitatively compare their biogeochemical model results with a wide range of observations. This comparison will be made in an open way that is transparent and understandable to the entire community, and thereby facilitate more rapid improvement of the biogeochemistry component of CCSM.

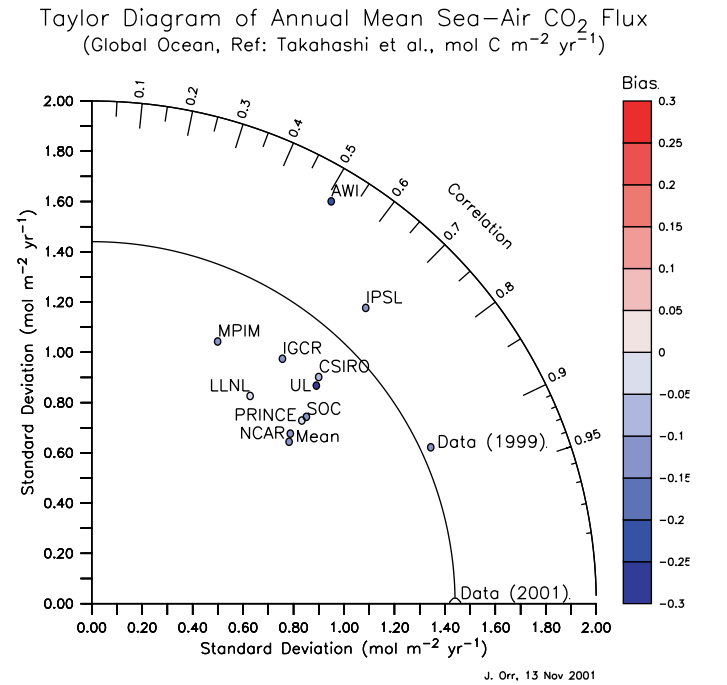


Fig. 41. "Taylor diagram" comparing OCMIP2 results (including those from NCAR) with air-sea CO_2 fluxes inferred from observations. NCAR's model performs among the best of all modeling groups on this measure.

Progress

Under SCIDAC funding,

- We have developed tools to apply a single analysis to all OCMIP models at once on all OCMIP grids including the POP grid (see Figure 41).
- We can now plot all OCMIP models at once on all OCMIP grids including the POP grid.
- We have developed several biogeochemically-oriented diagnostics of model performance, and tools for handling data and calculating commonly needed quantitities (see Figure 42).
- We have begun development of the production automation system — the application framework that will automatically process model results and produce an analysis document.

We have met with our colleagues who are developing diagnostics for European coupled modeling efforts, and who are coordinating OCMIP3. As a result of these meetings, diagnostic capabilities developed in Europe can be incorporated into this SCIDAC effort. Furthermore, if successful, the tools developed under this SCIDAC funded work will contribute to the analysis of OCMIP3 results.

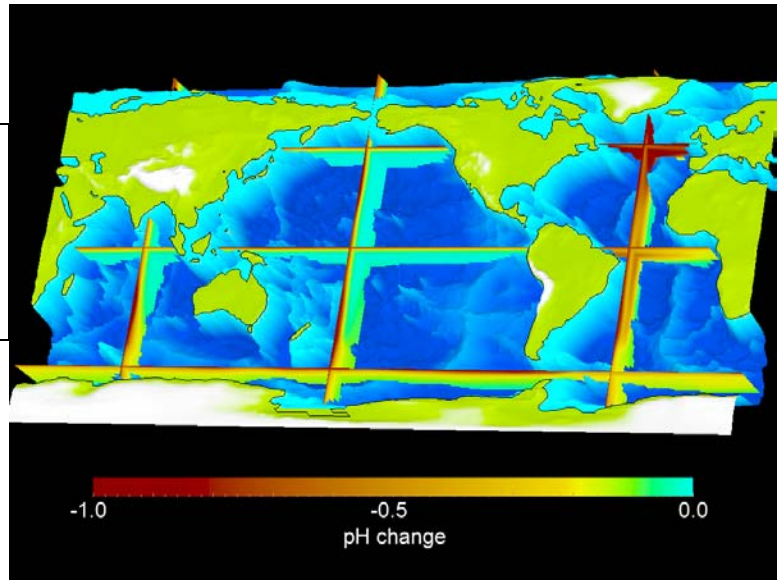


Fig. 42. Ocean pH changes computed for a model simulation for a business as usual scenario. These pH changes were calculated using diagnostic tools developed in this project.

Our current level is adequate to develop a useful ocean biogeochemistry diagnostic framework that can be applied to CCSM within the 5 year period. With a ramp up of funds we could expand this effort, as well as meet our originally proposed objectives of developing, an improved, more mechanistic representation of remineralization in the deep ocean and recycling of trace nutrients in the upper ocean.

We have completed tasks listed above and we are on schedule to have an up-and-running ocean biogeochemistry diagnostic system suitable for use with CCSM.

Plans

The main priority is to get an ocean biogeochemistry diagnostic system suitable for use with CCSM up and running on a server.

The next priority is to begin using this system to diagnose CCSM results, and through an iterative process, add additional useful diagnostics to the system.

We need to coordinate with those running the CCSM biogeochemistry model so that we can use their model output in development of the diagnostics system. Our current testing regime includes analysis of NCAR's submission to the Ocean Carbon Model Intercomparison Project. The system developed under this project is coordinated with OCMIP efforts and a NASA funded project to compare satellite observations with model predictions of ocean color.

11.0 COORDINATION AND MANAGEMENT (R. MALONE AND J. DRAKE)

11.1 MANAGEMENT PLAN

In a collaboration such as this, which involves eight major research institutions, it is imperative that roles and responsibilities be clearly defined. Toward this end, a Management Plan[a] (MP) was produced and reviewed by at least the lead investigator at each laboratory. The MP defines key topical areas in which this SciDAC project intends to contribute to CCSM. Each DOE laboratory has decided to which areas it is qualified to contribute and has selected a "Topic Leader" in each such area to oversee that laboratory's activities and serve as a contact point in coordinating with other institutions in the Consortium. The PIs have appointed one such Topic Leader to be the consortium Topic Coordinator; it is the responsibility of this person to coordinate activities among the six DOE laboratories and with the NCAR counterpart in the given topic area. The current assignments are summarized in Table 8.

The MP indicates the mechanisms by which this Consortium project will coordinate its activities with those of others contributing to the development and validation of the CCSM. The first mechanism is the management hierarchy of the CCSM projects itself, funded by the National Science Foundation (NSF): the Working Groups, the Scientific Steering Committee, and the CCSM Advisory Board. Second, the DOE CCPP continues to make important contributions to CCSM. More recently, the NASA-sponsored ESMF project has come into existence. Finally, many other aspects of the SciDAC program are relevant to CCSM. The MP addresses coordination with all these. Ultimately, it is collaboration at the working level that determines the success of the relationship, not managerial pontification. In that regard, it is essential that everyone involved in the Consortium participates actively in the CCSM Working Groups and attends the annual summer CCSM Workshop. This close contact assures that SciDAC is working on relevant topics and is in sync with CCSM activities.

From our vantage point, all participating organizations are cooperating harmoniously. We believe that, with the support of SciDAC management, this can be sustained. To the best of our knowledge, each organization is contributing in a timely fashion and otherwise fulfilling its role(s). We are in frequent contact with our colleagues at NCAR, and we believe that they firmly believe that the contributions of the Consortium have greatly accelerated the progress of the CCSM.

Because the Consortium is part of a much larger SciDAC program, there are many other activities that are relevant to the work that the Consortium does. The SciDAC ISICs and Collaboratories that have the greatest potential impact on CCSM are listed in Table 9, along with the names of the principal investigators and key laboratory personnel.

[a] The full text of the management plan was published in an appendix of the May 2002 progress report. A copy of the Management Plan appears in Appendix A for the benefit of the committee members at the October 2003 review.

Table 8. Laboratories participating in the SciDAC Consortium project and the Topics to which each is contributing. The names of the Topic Leader at each lab in the indicated topic are listed. Names in **bold** have been designated Topic Coordinators; they are responsible for coordinating the contributions of all labs working on that topic. Names in *italics* indicate lead collaborators who are *not* currently funded by the SciDAC Consortium project.

Topic	NCAR	DAO	ANL	LANL	LLBL	LLNL	ORNL	PPNL
Site co-PIs	***CRAIG***	***LIN***	Taylor	MALONE	Ding	Cameron-Smith	DRAKE	Ghan
Coupler	***CRAIG***		LARSON	Jones	Ding			
Atmosphere	***HACK***	***LIN***				Mirin	DRAKE	Ghan
Ocean	***GENT***			JONES		Duffy		
Sea ice	***GENT***			LIPSCOMB				
Land surface	***BONAN***						HOFFMAN	
Software eng.	***CRAIG***	***PUTMAN***	Jacob	JONES	Ding	Mirin	Worley	
Atmos. chem.	Lamarque		Taylor			CAMERON-SMITH	Erickson	
Biogeochem.				Chu	Wehner	***THOMPSON***	ERICKSON	

Table 9. SciDAC Consortium project interactions with SciDAC ISICs and Collaboratories and with DOE climate modeling projects.

Topic	OTHER	PI	NCAR	ANL	LANL	LBNL	LLNL	ORNL
SciDAC ISICs and Collaboratories								
Performance Evaluation (PERC)						BAILEY		Worley
Earth System Grid (ESG)			MIDDLETON	Foster			Williams	
Scientific Data Management (SDM)			Middleton	Gropp		SHOSHONI		Burris
Common Component Arch (CCA)	Sandia/LLNL	ARMSTRONG	N. Collins	Larson	Rasmussen			Ham
Tera-scale Sim Tools & Tech (TSTT)	Stony Brook	GLIMM					Brown	
DOE climate modeling projects								
Climate model on a geodesic grid	Colorado St.	RANDALL			Baumgardner			
Climate Change Prediction (CCPP)			Washington		Malone		**BADER**	

11.2 COORDINATION WITH SCIDAC INTEGRATED SOFTWARE INFRASTRUCTURE CENTERS AND COLLABORATORIES

Collaboration with the SciDAC PERC has been very important. PERC has been aiding in benchmarking, performance instrumentation, and performance diagnosis. PERC collaborators have created a performance model for POP and this will aid in future performance improvements. Collaborations with PERC and with the CCSM Software Engineering Group have been helpful in porting and determining optimal tuning parameter settings for CAM on a number of different platforms. We have also been tracking the performance evolution of CAM and identifying remaining performance problems.

ESG is collaborating on the rapid transfer of model output between computing and analysis centers. We can report that a number of prototype tools have been tested including *gridftp*, with varying degrees of success. The availability of production quality tools appears to be hampered by firewalls, rapidly changing security procedures and operational differences among the centers.

The Scientific Data Management (SDM) Center has also become actively involved with climate data concerns. Discussions on a broad range of issues including analysis methods for ongoing simulations, factor analysis, metadata and hierarchical data managers are ongoing. Two particular topics have resulted in focused collaborations. First, a parallel netCDF library is being developed at Northwestern University and Argonne National Laboratory under SciDAC for use in the distributed memory computers. LBNL provided a comprehensive parallel netCDF I/O test code that can do tests on arbitrary parallel domain decomposition on a 3-D array. The test code is part of the standard test for the NU/ANL parallel netCDF releases. LBNL has also tested the NU/ANL codes and sent their results to the SDM project. LBNL is studying the interface and plan to integrate it into their ZioLib for implementation into CAM and other CCSM components. NetCDF is a self-describing data format that has become the *de facto* standard in the US climate community and in particular, is the data format implemented in the CCSM code. Second, some climate analysis tools have been modified to allow direct access into HPSS archives. This eliminates one step in a climate scientist's data management practice. This ability has been prototyped in *ncview* and plans are to also modify the GRADS tool and the PCMDI tools.

In the area of software engineering and "performance portability" (the ability to maintain high performance on multiple computer architectures with a single version of a code), three projects are working toward similar goals:

1. The MCT developed at ANL as part of this SciDAC Consortium project. MCT has been employed to create a new, much more flexible and efficient flux coupler, CPL6, for CCSM. CPL6 has been implemented, tested, and accepted for use in the current version of CCSM.
2. NCAR is the lead laboratory for the development of the NASA-funded ESMF program. For several decades, the weather/climate community has struggled with the issues of interoperability and reuse of software. The goal has been to define coding and interface standards that, if widely adopted, would ease the interchange of model components developed at different organizations. ESMF represents the first major step in this direction that is likely to be widely adopted within the geophysical research community.

If ESMF meets its objectives, an ESMF-based framework will replace the MCT-based CPL6.

3. The third activity is the DOE CCA program. Like ESMF, this program has the objective of creating a suite of software components that can greatly reduce the difficulty of porting codes across different architectures by encapsulating the architecture-dependent features in the CCA (or ESMF) components. CCA is intended to be applicable to a wide range of scientific codes, while ESMF will be designed and optimized specifically for climate and other geophysical codes.

A difficult issue arose between SciDAC and ESMF, regarding perceived competition between ESMF and the DOE/SciDAC-sponsored MCT. This was dealt with by developing a Memorandum of Understanding[b] (MOU) between ESMF and the SciDAC Consortium project. Appropriate levels of management on both sides have approved the MOU.

11.3 COORDINATION WITH OTHER DOE/SCIDAC CLIMATE MODELING PROGRAMS

This project is complementary to the DOE CCPP. The CCPP has four components that emphasize physical aspects of climate model development, validation, and application. These are embodied in the Climate, Ocean, and Sea Ice Modeling (COSIM) project at LANL, the Program for Climate Model Diagnosis and Intercomparison (PCMDI) at Livermore, the climate modeling project headed by Warren Washington at NCAR, and a set of grants to university scientists that spans other aspects of climate modeling and research. The SciDAC Consortium project is focused on numerical methods, performance optimization (with emphasis on dynamical cores), portability of model codes across a wide range of computer architectures, and addition to the CCSM of atmospheric chemistry and ocean biogeochemistry. The SciDAC Consortium is CCSM-centric; CCPP is not, although many aspects of CCPP do pertain to the CCSM.

The Consortium Project also has substantial overlap with the Colorado State University SciDAC project to build a geodesic-grid climate model. It will use versions of the Los Alamos HYPOP and CICE models that have been modified to work on geodesic grids.

11.4 CURRENT TASKS AND LEVELS OF EFFORT

There are three FTEs at NCAR paid through SciDAC funds. These include two software engineers in the CGD division and one scientist in the ACD division. The two software engineers in CGD were hired in the spring of 2002. They are part of the CCSM Software Engineering Group. This group provides core support for model development and other activities within CCSM.

One of the FTEs in CGD, Wei Yu, is taking a lead in testing activities. He carries out regular tests of new versions of CCSM, verifies changes in the model, and conducts weekly tests of release versions on supported platforms. Regular and formal testing of CCSM on multiple platforms, in different configurations and at different resolutions is critical. Having a full-time test engineer has significantly improved the robustness of model changes and releases and has also decreased the down time due to

[b] The full text of the SciDAC-ESMF MOU was published in an appendix of the May 2002 progress report.

model bugs because they are detected earlier. Wei is also playing a large role in developing a new user GUI for CCSM, and he also carries out model development activities as time allows.

The other FTE in CGD, Dan Miner, is taking a lead managing and improving the CCSM software infrastructure. Currently, he is charged with managing the CVS repository and all associated infrastructure including help pages, providing new user access, security, and support scripts. He has also written scripts in support of the CCSM release web pages and is testing a new bug tracking tool that CCSM expects to start using soon. In the future, Dan will likely continue to bring new infrastructure tools on-line, and in the long term, we hope to build an infrastructure that brings together web pages, user GUIs, databases, source code control, and bug tracking to improve communication and coordination of CCSM model development activities and experiments.

The scientist in ACD, Jean Francois Lamarque, is currently developing the chemical schemes to be included in CAM2. This new capability will allow users to study the coupled atmosphere-chemistry system in the context of climate studies. Several schemes with various degrees of complexity in the chemical system are being developed. This on-going effort will use available observations of chemical compounds to validate the schemes. This coupled model will then be used for the study of the interactions between the atmospheric chemistry and the climate under past and future conditions.

The following tables indicate how the SciDAC resources at each Laboratory have been allocated during the time interval covered by this progress report. "FTE" is the level of effort in units of fraction of the person's time. The general task area(s) are indicated, followed by the section in which the details of progress are given.

Argonne National Laboratory

Person	**FTE**	**Current tasks**
J. Taylor	0.9	Coordination and management Biogeochemistry Atmospheric Chemistry
J. Larson	0.75	Coupler Development Coordination and management
R. Jacob	0.75	Coupler Development Coordination and management
E. Ong	0.75	Coupler Development
Total FTEs	3.15	
Annual budget	$700K	@ average annual cost per FTE = $220K

Lawrence Berkeley National Laboratory

Person	**FTE**	**Current tasks**
C. Ding	0.40	I/O, Coupler, and Coordination
Y. He	0.60	Coupler (MPH), Pilgrim, OpenMP
W. Yang	1.00	I/O library
Total FTEs	2.00	
Annual budget	$400K	@ average annual cost per FTE = $200K

Lawrence Livermore National Laboratory

Person	**FTE**	**Current tasks**
D. Rotman	0.20	Coordination, Atmospheric Chemistry
P. Connell	0.20	Atmospheric Chemistry, mechanism definition
A. Mirin	0.40	Software Engineering, optimization of Lin-Rood

P. Cameron-Smith	0.60	Atmospheric Chemistry, radiation balance
J. Tannahill	0.30	Software Engineering, atmospheric chemistry
K. Grant	0.20	Radiative Forcing
P. Duffy	0.20	High Resolution climate modeling
M. Wickett	0.20	Ocean Biogeochemistry
B. Govindasamy	0.20	High Resolution climate modeling
Total FTEs	2.5	
Annual budget	$800K	@ average annual cost per FTE = $320K

Los Alamos National Laboratory

Person	**FTE**	**Current tasks**
R. Malone	0.40	Coordination and management
J. Dukowicz	0.40	HYPOP development
B. Lipscomb	0.30	HYPOP , CICE performance
J. Baumgardner	0.60	HYPOP development
P. Jones	0.30	SWE and performance
R. Smith	0.35	HYPOP development
S. Elliot	0.25	Biogeochemistry
S. Chu	0.50	Biogeochemistry
Total FTEs	3.10	
Annual budget	$800K	@ average annual cost per FTE = $258K

National Center for Atmospheric Research

Person	**FTE**	**Current tasks**
W. Yu	1.0	SWE, Testing, Performance Assessment and Optimization
D. Miner	1.0	SWE, Infrastructure Support and Development, Repository Management
J.F. Lamarque	1.0	Atmospheric Chemistry
Total FTEs	3.0	
Annual budget	$600K	@ average annual cost per FTE = $200K

Oak Ridge National Laboratory

Person	**FTE**	**Current tasks**
J. Drake	0.80	Coordination and management Atmospheric Model Land Surface Model
P. Worley	0.50	Software Engineering Atmospheric Model
D. Erickson	0.50	Biogeochemistry Atmospheric Chemistry
F. Hoffman	0.50	Land Surface Model
M. Ham	0.30	Software Engineering
M. Branstetter	1.00	Land Surface and River Routing
Total FTEs	3.60	
Annual budget	$850K	@ average annual cost per FTE = $236K

Pacific Northwest National Laboratory

Person	**FTE**	**Current tasks**
S. Ghan	0.33	Subgrid orography scheme
T. Shippert	0.33	Subgrid orography scheme
Total FTEs	0.66	
Annual budget	$150K	@ average annual cost per FTE = $230K

PUBLICATIONS DURING 2001-2003

M. L. Branstetter and D. J. Erickson III, "Continental runoff dynamics in the CCSM2.0," J. Geophys. Res, Vol. 108, NO. D17, 4550, doi:10.1029/202JD003212, 2003.

S. Chu, S. Elliott, M. Multrud, J. Hernandez and D. J. Erickson III, "A global eddy resolving simulation of oceanic DMS production and air-sea flux," In prep., 2003.

C. Ding and Y. He, "A ghost cell expansion method for reducing communications in solving PDE problems," Proceedings of Supercomputing 2001, 2001.

D. J. Erickson, III and J. L. Hernandez, "A global, high resolution, satellite-based model of air-sea isoprene flux," In *Gas transfer at water surfaces*, American Geophysical Union Monograph, (Eds., M. A. Donelan, W. M. Drennan , E. S. Saltzman and R. Wanninkhof), 127, 312-317, 2002.

D. J. Erickson, J. Hernandez, P. Ginoux, W. Gregg, C. McClain and J. Christian, "Atmospheric Iron Delivery and surface ocean biological activity in the Southern Ocean and Patagonian region," Geophys. Res. Lett., VOL. 30, NO. 12, 1609, doi:10.1029/2003GL017241, 2003.

S. J. Ghan, X. Bian, A. G. Hunt, and A. Coleman, "2002: The thermodynamic influence of subgrid orography in a global climate model," *Climate Dynamics*, 10.1007/s00382-002-0257-5, 2002.

Y. He and C. Ding, "MPI and OpenMP paradigms on cluster of SMP architecture: the Vacancy Tracking Algorithm for multi-dimensional array transpose," Proceedings of Supercomputing 2002.

J. L. Hernandez and D. J. Erickson III, "Modulation of the Surface Energy Budget by atmospheric dust in the Western Tropical Atlantic: A 17 Year Analysis," Submitted, J. Geophys. Res, 2003.

W. H. Lipscomb and E. C. Hunke, "Modeling sea ice transport using incremental remapping." Submitted to Monthly Weather Review, 2002.

E. T. Ong, J. W. Larson, and R. L. Jacob, "A Real Application of the Model Coupling Toolkit," To appear in Proceedings of the 2002 International Conference on Computational Science (ICCS), to be published by Springer-Verlag, 2002.

R. J. Oglesby, S. Marshall, D. J. Erickson III, J. O. Roads and F. R. Robertson, 'Thresholds in atmosphere-soil moisture interactions: Results from climate model studies," J. Geophys. Res., 107, D14, 10.10294/20015D001045, 2002.

S. A. Mickelson, J.A. Taylor, and M. Dvorak (2002). "Simplifying the Task of Generating Climate Simulations and Visualizations," Proc. 2002 International Conference on Computational Science, eds. Peter M.A. Sloot, C.J. Kenneth Tan, Jack J. Dongarra, Alfons G. Hoekstra, Springer-Verlag.

J. A. Taylor, M. Dvorak, S.A. Mickelson (2002), "Developing Grid Based Infrastructure for Climate Modeling," Proc. 2002 International Conference on Computational Science, eds. Peter M.A. Sloot, C.J. Kenneth Tan, Jack J. Dongarra, Alfons G. Hoekstra, Springer-Verlag .

Rotman D.A., C.S. Atherton, D.J. Bergmann, P.J. Cameron-Smith, C. C. Chuang, P.S. Connell, J.E. Dignon, A. Franz, K.E. Grant, D.E. Kinnison*, C.R. Molenkamp, D.D. Proctor, J.R. Tannahill, "IMPACT, the LLNL 3-D global atmospheric chemical transport model for the combined troposphere

and stratosphere: Model description and analysis of ozone and other trace gases," *J. Geophys. Res.*, in press, 2003.

J. S. Vetta and P. H. Worley, "Asserting Performance Expectations," Proceedings of the ACM/IEEE Conference on High Performance Networking and Computing (SC02), Baltimore, MD, November 16, 2002.

P. H. Worley, "Scaling the Unscalable: A Case Study on the AlphaServer SC," Proceedings of the ACM/IEEE Conference on High Performance Networking and Computing (SC02), Baltimore, MD, November 16, 2002.

W-S. Yang and C. Ding, 2003, ZioLib: a Parallel I/O Library, LBNL Tech. Report, LBNL-53521.

References:

[1] Collins, W.D., Hack, J.J., Boville, B.A., Rash, P.J., Williamson, D.L., Kiehl, J.T., Briegleb, B., McCaa, J.R., Bitz, C., Lin, S.J., Rood, R.B., Zhang, M., Dai, Y., Description of the NCAR Community Atmosphere Model (CAM2)

[2] Atmospheric Model Intercomparison Project (AMIP2) Sea Surface Temperature and Sea Ice data, http://www-pcmdi.llnl.gov/amip/AMIP2EXPDSN/BCS/bcsintro.html

[3] Observational reference datasets used by the Program for Climate Model Diagnosis and Intercomparison (PCMDI) for Taylor diagrams, http://esg.llnl.gov/AMWG/list_of_obs_datasets.html

[4] Climate Prediction Center Merged Analysis of Precipitation (CMAP), http://www.cpc.ncep.noaa.gov/products/global_precip/html/wpage.cmap.html

[5] NCEP /DOE AMIP-II Reanalysis, http://wesley.wwb.noaa.gov/reanalysis2/index.html

[6] Cess, R. D., and Zhang, M., Cloud structure anomalies over the tropical Pacific during the 1997/98 El Nino. *Geophysical Research Letters*, **28**, 4547-4550, 2001.

[7] Jianwei Li, Wei-keng Liao, Alok Choudhary, Robert Ross, Rajeev Thakur, William Gropp, and Rob Latham. "Parallel netCDF: A Scientific High-Performance I/O Interface," accepted by 15th Supercomputing conference, November 2003

[8] Leung, L. R., and S. J. Ghan, 1995: A subgrid parameterization of orographic precipitation. *Theoretical and Applied Climatology*, *52*, 95-118.

[9] Leung, L. R., and S. J. Ghan, 1998: Parameterizing subgrid orographic precipitation and surface cover in climate models. *Mon. Wea. Rev.*, *126*, 3271-3291.

[10] CCSM Scientific Steering Committee, "Community Climate System Model Plan (2000-2005)," June 2000.

[11] CCSM Scientific Steering Committee, "Community Climate System Model Science Plan (2004-2008)," June 2003.

[12] CCSM Scientific Steering Committee, "Community Climate System Model Strategic Business Plan (2004-2008)," June 2003.

[13] James Hansen, Makiko Sato, Reto Ruedy, Andrew Lacis, and Valdar Oinas, "Global warming in the twenty-first century: An alternative scenario," Proc Natl. Acad. Sci. USA, 10.1073/pnas.170278997.

[14] IPCC, 2001: "Climate Change 2001: The Scientific Basis. Contribution of Working Group I to the Third Assessment Report of the Intergovernmental Panel on Climate Change," edited by Houghton, J.T., Y. Ding, D.J. Griggs, M. Noguer, P.J. nan der Linden, X. Dai, K. Maskell, and C.A. Johnson. Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA, 881pp.

12 APPENDIX A: MANAGEMENT PLAN

12.1 INTRODUCTION[†]

This project employs multi-disciplinary teams to accelerate development of the Community Climate System Model (CCSM), based at the National Center for Atmospheric Research (NCAR). A consortium of six Department of Energy (DOE) National Laboratories will collaborate with NCAR and the National Aeronautics and Space Administration (NASA) Data Assimilation Office (DAO). The laboratories are Argonne (ANL), Los Alamos (LANL), Lawrence Berkeley (LBNL), Lawrence Livermore (LLNL), Oak Ridge (ORNL), and Pacific Northwest (PNNL). The proposed work focuses on software design with extensive documentation of the CCSM and its component models; performance optimization of the dynamical cores and other critical aspects of the atmosphere, ocean, and sea ice models; and the introduction of atmospheric chemistry and ocean biogeochemistry into the model.

Key objectives are to develop, validate (through comparison with observed data), document and optimize the performance of the CCSM, a comprehensive coupled climate model, using the latest software engineering methodology. This management plan defines roles and responsibilities to assure that the proposed work remains coordinated, focused, and compatible with the objectives of the CCSM Scientific Steering Committee, while supporting the DOE Climate Change Prediction Program (CCPP).

As part of the DOE Scientific Discovery through Advanced Computing (SciDAC) program, the "CCSM Consortium" must also work collectively with applicable parts of the whole SciDAC infrastructure development efforts, namely the Integrated Software Infrastructure Centers (ISICs) and National Collaboratories (NCs). The project also acts as a focal point for collaborations with related efforts sponsored by other agencies, such as the NASA Earth System Modeling Framework (ESMF).

12.2 OBJECTIVES

The overarching objective of the SciDAC CCSM Model Development project is to work in collaboration with NCAR to develop and maintain the CCSM as a state-of-the-art climate model optimized for performance, portability, and interoperability on a range of parallel computer architectures. Another objective is to facilitate its use to gain the best possible scientific understanding of climate variability and global change on decadal to century time scales. This project seeks to implement modern software engineering practices and modular, open development of each component of the CCSM. Efficient parallel execution for high-throughput climate simulations at multiple resolutions will be achieved through flexible model configurations and optimized utilities and algorithm libraries. The completeness of the model will also be extended through the development of new physical parameterizations, chemical and biogeochemical process models, more accurate dynamical representations, and more efficient solution methods.

12.3 MULTIPLE LEVELS OF COORDINATION[*]

Coordination of the SciDAC project must take place at several levels:

- It is necessary to coordinate contributions from several DOE labs in each of many topic areas (see Table 1 below).

[†] A complete list of acronyms is given in Appendix A.

[*] Full names, affiliations, and primary interests and roles of individuals referred to in this document are listed in Appendix B.

- Activities in DOE labs must be coordinated with related activities at NCAR and NASA/DAO, to avoid duplication of effort and to insure that DOE contributions are both timely and relevant to the overall CCSM goals.
- SciDAC activities ultimately must be compatible with the overall vision for CCSM determined by the collection of CCSM Working Groups (WGs) and the CCSM Scientific Steering Committee (SSC).
- SciDAC activities must complement, yet be compatible with, CCPP objectives.
- As part of the DOE SciDAC program, this SciDAC Consortium project is expected to interact with and benefit from research and development activities in the rest of the SciDAC program, namely, the ISICs and NCs.
- SciDAC activities must be compatible with NCAR plans for future evolution of CCSM, for example, the expected adoption of ESMF.

Clearly, for DOE laboratory scientists to be effectively involved in so many aspects of CCSM, it is imperative that they participate actively in relevant activities. Most important is involvement in the Working Groups as members or, if possible, as co-chairs. This is where alternative approaches are evaluated and compared prior to making recommendations to the SSC. Attendance at the annual summer meeting in Breckenridge provides a unique opportunity to see the "big picture" of CCSM's progress. Obviously, if the opportunity arises, it can be very helpful to serve as a member of the CCSM Advisory Board (CAB) or SSC.

Coordination among DOE Labs

This SciDAC Consortium project involves different subsets of six DOE laboratories contributing to different aspects of CCSM development and evaluation (see Table 1). The nomenclature used to describe different managerial responsibilities is explained here. Coordination will take place by means of weekly teleconference calls, supplemented by monthly Access Grid meetings. Problems will dealt with via email, telephone and mini-teleconference calls. Semi-annual meetings will be held. One will take place In Breckenridge on the Monday immediately preceding the annual CCSM Workshop in late June. The second will be held in conjunction with the periodic CCPP program review, if conveniently spaced in time, in conjunction with a major SciDAC meeting, or possibly via the Access Grid. A project web page is maintained at http://www.scidac.org/CCSM.

Principal Investigators (PIs)

The PIs, Bob Malone, (LANL) and John Drake, (ORNL), will coordinate the all aspects of the project among the participating DOE laboratories and corresponding activities at NCAR and NASA/DAO. The PIs are responsible for: (a) monitoring progress of SciDAC tasks, (b) negotiating the roles of the DOE labs relative to one another and NCAR, and (c) insuring through discussions with CCSM management that SciDAC contributions are compatible with CCSM objectives. The PIs will provide semi-annual reports on the progress of the project to the DOE program sponsors. These will include highlights of the project and status of scheduled tasks and milestones. Drake and Malone will also act as serve as liaisons to NCAR and CCSM management for the coordination of the project.

Laboratory Coordinators

Each participating DOE lab has a designated Laboratory Coordinator for the project, whose purview cuts across all topics being pursed at that laboratory. It is the responsibility of the Lab Coordinators to communicate issues and progress to the PIs and to help manage ongoing activities at their labs. Laboratory Coordinators and their NCAR and NASA/DAO counterparts are listed (italicized) in the

second row of Table 1. Generally speaking, the Lab Coordinators are the PIs and co-PIs listed on the title page.

Topic	NCAR	NASA/ DAO	ANL	LANL	LBNL	LLNL	ORNL	PNNL
Coordinators	*Collins*	*Lin*	*Taylor*	***Malone***	*Ding*	*Cameron-Smith*	***Drake***	*Ghan*
SWE	Craig	Sawyer	Larson	**Jones**		Mirin	Worley	
CAM	Hack	Lin	Jacob		Yang	Mirin	**Drake**	Ghan
COM/POP	Gent		Jacob	**Jones**		Duffy		
CSIM/CICE	Holland		Taylor	**Lipscomb**				
CLM	Bonan		Larson				**Hoffman**	Ghan
Coupler	Craig		**Larson**	Jones	Ding			
Atmos Chem	McKenna		Taylor			**Cameron-Smith**	Erickson	
Biogeochem	Doney		Taylor	Maltrud		Thompson	**Erickson**	

Table 1. Laboratory Coordinators (italicized in second row) are responsible for overseeing all activities at their respective institutions and high-level coordination among institutions. Major topic areas (component models, major new subcomponent models) are listed in the first column, followed by Topic Leaders and (in bold) Topic Coordinators.

Topic Leaders

Each lab that is contributing to a particular topic area has selected a "Topic Leader," whose job is to monitor progress within that lab and coordinate with Topic Leaders for the same topic at other DOE labs.

Topic Coordinators

In each topic, the PIs have appointed one of the Topic Leaders the overall DOE lab "Topic Coordinator," whose job is to monitor progress and, with the help of the Topic Leaders, coordinate work among the DOE labs and with NCAR and NASA/DAO.

Coordination with CCSM

The management structure of the CCSM project at NCAR has three parts: the SSC, the CAB, and a collection of WGs, each devoted to a component model, scientific research area, or other aspect of CCSM. Detailed information about the CCSM management structure and scientific plans is available on-line at http://www.cgd.ucar.edu/csm.

Working Groups

Each of the topic areas listed in Table 1 falls within the scope of some CCSM WG. Each working group consists of scientists who come together to work on topics in which they share common interest. Membership in any WG is open to all persons having an interest in the topic. The WGs allow scientists to participate in cooperative research, compare different approaches, and minimize unnecessary duplication. The WGs present their research and recommendations, preferably based on consensus, to the SSC, which

has the authority to accept or reject any recommendation. The SSC may also call for further research before any decision is made. Thus, it is imperative that SciDAC personnel be involved closely in the activities of relevant WGs.

At present, there are nine WGs, each co-chaired by one or more non-NCAR scientists and zero or more NCAR-based scientists. The topic areas and co-chairs of each WG are displayed in Table 2. DOE laboratory scientists appear in italics.

Working Group	Co-chairs	Affiliations
Atmosphere Model	Phil Rasch	NCAR
	Leo Donner	GFDL
Ocean Model	Bill Large	NCAR
	Richard Smith	*Los Alamos National Laboratory*
Land Model	Gordon Bonan	NCAR
	Steven Running	University of Montana
Polar Climate	Richard E. Moritz	University of Washington
	Elizabeth Hunke	*Los Alamos National Laboratory*
Paleoclimate	Bette Otto-Bliesner	NCAR
	Lisa Sloan	University of California-Santa Cruz
Climate Variability	Mike Alexander	NOAA/CDC
	Clara Deser	NCAR
Biogeochemistry	Scott Doney	NCAR
	Inez Fung	University of California-Berkeley
Climate Change And Assessment	Warren Washington	NCAR
	Gerald Meehl	NCAR
	Ben Santer	*Lawrence Livermore National Laboratory*
Software Engineering	*Pat Worley*	*Oak Ridge National Laboratory*
	Cecelia DeLuca	NCAR
	Tony Craig	NCAR

Table 2. Names (topic areas) of CCSM Working Groups and Co-chairs.

Communications with the WGs occurs only through participation by DOE scientists and managers in relevant WGs. In several cases, DOE scientists are co-chairs of WGs. Active involvement is the most effective way to influence the direction taken by WGs.

Scientific Steering Committee

The CCSM SSC provides scientific leadership for the CCSM project, including oversight of activities of working groups, coordination of model experiments, decision making on model definition and development, and encouragement of external participation in the project. The SSC determines what working groups should be organized and oversees the activities of these working groups. The co-chairs for each working group are appointed by the SSC. The major scientific responsibility of the SSC is to decide which components and/or parameterizations should be included in future versions of CCSM. Proposals for new components and/or parameterizations should come from the appropriate working groups, together with appropriate reasons for the recommended changes and documentation of the results.

Bill Collins, Chair	NCAR
Chris Bretherton	University of Washington
Ping Chang	Texas A&M University
Jim Hack	NCAR
Bill Large	NCAR
Maurice Blackmon	NCAR
Cecilia Bitz	University of Washington
Daniel McKenna	NCAR
Scott Doney	Woods Hole
Gordon Bonan	NCAR
Ben Santer	LLNL

Table 3. Membership of the CCSM Scientific Steering Committee.

The CCSM SSC members consist of the Director of NCAR's Climate and Global Dynamics (CGD) Division plus eight additional scientists. The present membership of the SSC is given in Table 3.

It is worth noting that most SciDAC scientists have no access to the SSC as a unit, so the only mechanism that presently exists for communication between SciDAC and the SSC is conversations with individual members of the SSC. That seems sufficient for now.

CCSM Advisory Board

In addition to the SSC, the CAB meets twice annually to review the progress and status of the CCSM program. The CAB then writes a report (letter) to the President of University Corporation for Atmospheric Research, the Director of NCAR, and the Leader of the CGD Division. In January 2002, John Drake became a member of CAB, taking over from Bob Malone, who served a three-year term starting in 1998.

Coordination with CCPP

In its present form, the CCPP comprises numerous university grants plus three major projects: (1) the Parallel Climate Model (PCM) project at NCAR, (Warren Washington, PI); (2) the Program for Climate Model Diagnosis and Intercomparison (PCMDI) at LLNL, (Doug Rotman, Acting Director); and (3) the Climate, Ocean and Sea Ice Modeling (COSIM) project at LANL, (Bob Malone, PI). PCM is merging with CCSM2 in 2002, while the COSIM project supplies the Parallel Ocean Program (POP) and much of the sea ice component model (CICE) to the CCSM2. PCMDI is playing an important role in evaluation and validation of CCSM2. Thus strong ties already exist between CCPP, CCSM, and the SciDAC CCSM Consortium.

Coordination with SciDAC Program Elements

The full SciDAC program within DOE spans a wide range of applications, of which climate modeling is the largest. Cross-cutting activities in numerical methods, adaptive grids, mathematical libraries, data management, and computational performance optimization are supported by seven ISICs. Software tools that support multi-site interactions of various types are supported by ten NCs projects. Collaborations between the SciDAC CCSM Consortium and several of the SciDAC ISICs and Collaboratories have been established. Table 4 lists those most pertinent to the SciDAC CCSM Consortium project, along with the lead investigators at each institution; the Principal Investigator of each ISIC or NC is in bold. Tasks in collaboration with other SciDAC projects are tracked along with other project tasks. Note that the

Geodesic Climate Model project led by David Randall (CSU) is another SciDAC Climate project, not an ISIC or NC. The Earth System Grid is an NC.

ISIC/NC	NCAR	ANL	LANL	LBNL	LLNL	ORNL	Other
Earth Sys Grid	Middleton	**Foster**			Williams	White	
Geodesic CM			Baumgardner				**Randall** (CSU)
PERC ISIC				**Bailey**		Worley	
CCA ISIC		Freitag	Rasmussen		**Armstrong**	Bernholdt	
TSTT ISIC		Freitag			Brown		**Glimm** (BNL)
TOPS ISIC					**Keyes**		
APDEC ISIC				**Colella**	Brown		

Table 4. SciDAC projects pertinent to CCSM Consortium. PI names in bold.

Coordination with ESMF.

The ESMF project is a three-year, multi-institutional effort to develop a modeling framework that can be adopted by all of the major weather and climate modeling institutions in the US. It is funded by NASA as the third phase of its High Performance Computing and Communications Program. NCAR is the lead organization, and the fact that the lead PI is Tim Killeen, director of NCAR, indicates the strong commitment that NCAR has to make this project successful and to adopt ESMF in a future version of CCSM.

The ESMF will establish standards for interfaces between component models by means of a universal "coupler," interfaces to subcomponent models, grid types, field types, underlying utility and machine-dependent layers, and a control layer at the top. The objective is to greatly simplify the interchangeability of component models and subcomponent parameterizations while maintaining high performance and transparent portability across a range of computer architectures. Work funded by DOE can play an important role in ESMF. The Model Coupling Toolkit (MCT) developed at ANL under the Accelerated Climate Prediction Initiative "Avante Garde" pilot project and the present SciDAC CCSM Consortium, provides many of the foundational concepts for ESMF. The Spherical Coordinate Remapping and Interpolation Package (SCRIP) developed at LANL will also be adopted in modified form as part of ESMF's "toolkit." MCT is the basis for the newest coupler, CPL6, which will appear in a later version of CCSM2. CPL6 is also a prototype for the ESMF coupler.

Because the SciDAC Consortium project is focused exclusively on CCSM and its long-term development, and because NCAR has a strong commitment to adopt ESMF – if it meets its performance goals – SciDAC has an implied commitment to ESMF. While SciDAC funds will not be used to support ESMF directly, improvements in MCT and SCRIP under SciDAC will ultimately benefit ESMF. Also, Phil Jones, (LANL), Jay Larson and Rob Jacob, (ANL) are co-investigators on the ESMF proposal to NASA, and therefore stand to receive NASA funding to explicitly support their work on ESMF.

Communications with the ESMF project occurs directly through the ANL and LANL co-investigators named above, and frequent emails and telephone conversations with Cecilia DeLuca, (NCAR Co-PI on ESMF and Co-chair of the CCSM Software Engineering WG).

A Memorandum of Understanding between the NCAR-based ESMF project, the ANL-based MCT project, and SciDAC Consortium management has been written and signed.

13 APPENDIX B: NAMES, AFFILIATIONS, AND PRIMARY INTERESTS

The following table contains the full names, affiliations, and primary interests of individuals referred to in this document. Individuals whose names are italicized are funded by their own SciDAC projects, which are closely related to the present project.

Last name	First name	Laboratory	Primary Interest
Baumgardner	John	LANL	Ocean Model
Branstetter	Marcia	ORNL	River Runoff Model
Bettge	Tom	NCAR	Coupled Model
Bryan	Kirk	Princeton	Ocean Model
Buja	Lawrence	NCAR	SW Engineering
Cameron-Smith	Phillip	LLNL	Atmospheric Chemistry
Chu	Shaoping	LANL	Ocean Biogeochemistry
Craig	Tony	NCAR	SW Engineering
Ding	Chris	LBNL	SW Engineering
Drake	John	ORNL	Atmospheric Model
Duffy	Phil	LLNL	Hi-Res Atmosphere
Dukowicz	John	LANL	Ocean and Sea Ice Models
Elliott	Scott	LANL	Ocean Biogeochemistry
Erickson	David	ORNL	Biogeochemistry
Ghan	Steve	PNNL	Atmospheric Model
Ham	Michael	ORNL	SW Engineering
He	Yun	LBNL	SW Engineering
Hoffman	Forrest	ORNL	Land-Surface Model
Hunke	Elizabeth	LANL	Sea-Ice Model
Jacob	Rob	ANL	Coupler, SWE
Jones	Phil	LANL	Ocean Model
Lamarque	J.F.	NCAR	Atmospheric Chemistry
Larson	Jay	ANL	Coupler, SWE
Lin	S. J.	NASA/DAO	Lin-Rood Dycore
Lipscomb	Bill	LANL	Sea ice Model
Malone	Robert	LANL	Project Coordination
Maltrud	Matt	LANL	Ocean Analysis
McKenna	Daniel	NCAR	Atmospheric Chemistry
Mirin	Art	LLNL	Atmospheric Model
Putman	Bill	NASA/DAO	Atmospheric Model
Randall	David	CSU	Geodesic Grid Models
Rotman	Doug	LLNL	Atmospheric Chemistry
Sawyer	Will	NASA/DAO	SW Engineering
Schramm	Julie	NCAR	Sea Ice Model
Shippert	Tim	PNNL	Atmospheric Model
Smith	Rick	LANL	Ocean Model
Taylor	John	ANL	Biogeochemistry
Worley	Pat	ORNL	Model Performance
Yang	Woo-Sun	LBNL	SW Engineering