National Security policy-driven needs

US requires superiority in high end computing for:

- **Superiority in weapons design**
- **Comprehensive test ban treaty**
- **Critical reaction capability/time for defense**
- **Battlespace dominance; revolution in military affairs**

Superiority in critical defense technologies
requires
Superiority in high end computing technology

Computational Speedups of 1,000-1,000,000 required by 2010

- Nuclear weapons stockpile stewardship: 100,000 - 1,000,000X
- Wide area imagery: 2,000X
- Vehicle and weapons design and test
  - High performance aircraft design and test: 10,000-100,000X
  - Weapons systems such as high power RF weapons: 1,000+X
  - Target discrimination: 3,000
- Intelligence data and information extraction
  - Remote sensing exploitation: 10,000-100,000
  - Chemical detection (FTIR remote sensor): 300,000X
  - Intelligence data / Information extraction: 3,000X
- Synthetic theater of war C 3 : 100,000X

HECC R&D Scope

- System software and tools
- Application development environments
- Fast, efficient algorithms for simulation, modeling and visualization
- System architectures
- Device technologies
- Interconnection technologies
- I/O, and multi-level data storage
- Laboratory demonstration prototypes
- Advanced simulation of physical phenomena and other grand challenge applications
- High end computing systems serving as infrastructure for computational science and engineering research

Thrust 1: System Software Technology for High-End Computing

Metrics and performance goals (within 5 years):

- Scalability
  - logarithmic or better
- Speedup
  - 50% of ideal
- Portability
  - all major vendors
- Performance
  - 100-fold improvement in time-to-solution over FY96 baseline when supported by comparable advances in hardware and algorithm technologies

Thrust 1: System Software Technology for High-End Computing

1.1 Languages and compilers

1.2 Debugging and performance tools

1.3 Programming interfaces and libraries

1.4 Operating system services and I/O (large scale, highly parallel)

1.5 Common framework and infrastructure

1.1 Languages and compilers

Programmatic/Technical Objectives: Foster improved languages and compilers, which deliver to applications high performance, portability and enhanced scalability.

- Develop concurrency and data organization languages
- Implement parallelism and data organization compilers
- Implement dynamic load rebalancing runtime software
- Supply FORTRAN 90, HPF, and C++ on HPC platforms

1.2 Debugging and performance tools

Programmatic/Technical Objectives: Develop new generation debugging and tuning tools with a common look and feel for scalable parallel debugger and performance tool sets.

- Standardize HPC debugger command language and GUI
- Implement bug-free debuggers for SMP's
- Define parallel debugging functionality
- Develop techniques to debug state-of-interconnection networks
- Define/develop hardware performance counters and access API
- Develop multi-platform analysis tools for high performance architectures

1.3 Programming interfaces and libraries

Programmatic/Technical Objectives: Provide efficient, parallel libraries, with identical APIs on HPC platforms that improve HPC application support.

- Continue provision of HPC libraries of modules with domain of applicability and performance characteristics
- Develop application- or domain-specific environments
- Encourage interface and libraries usage

1.4 Operating system services and I/O (large scale, highly parallel)

Programmatic/Technical Objectives: Provide an efficient parallel operating system with common services across large-scale highly-parallel platforms.

- Wide implementation of explicit checkpoint/restarts
- Develop efficient schedulers
- Develop wide-area file systems
- Develop scalable resource management techniques

1.5 Common Framework and Infrastructure

Programmatic/Technical Objectives: Foster a hardware and software infrastructure for high-performance computing that enables HPC compilers, tools, and operating systems and provides a scalable common systems software framework.

- Develop HPC compiler infrastructure
- Develop HPC tools infrastructure (APIs, GUIs, program databases, and algorithms)
- Develop plug-and-play components
- Provide large-scale resources for O/S and tool development and R&D
- Establish a High Performance System Software Test and Evaluation Center
- Define common framework - API to machine interface
- Isolate architecture-dependent features

Thrust 2: Leading-edge research for future-generations computing Focus Areas

2.1 Innovative Technology

2.2 Systems and Architecture

2.3 Laboratory demonstration prototypes

2.1 Innovative Technology

Programmatic/Technical Objectives: Exploit innovative architectures and technology, including software and hardware components, to achieve PetaFLOPS-level computation and exabyte-level mass storage.

- Assess processor technologies, including superconducting rapid single flux quantum (RSFQ) devices
- Assess data communication technologies, including optical technology
- Assess memory and data storage technologies, including holographic devices
- Assess alternatives for a highly-scalable operating system kernel
- Alternative language paradigms for programming PetaFLOPS-class systems
- Potential new algorithms for key high-end applications

2.2 Systems and Architecture

Programmatic/Technical Objectives: Design systems and architectures including operating systems, compilers, programming languages, debuggers, visualization facilities, and performance tools to productively employ emerging future-generations computing power.

- Develop in greater detail hardware system architectures for future high-end systems
- Assess proposed systems performance using quantitative analyses
- Implement and test prototype operating system kernels on highly-parallel platform
- Develop in detail prototype programming language systems; assess effectiveness
- Develop prototype parallel performance, debugging and visualization tools on available hardware platforms
- Implement and test new algorithms for key applications; compare performance with more conventional methods

2.3 Laboratory demonstration prototypes

Programmatic/Technical Objectives: Deploy a specific hardware design, together with necessary components of system software for demonstration

- Engineer and manufacture one or more of the hardware designs, and deploy them in operational facilities available to application scientists
- Implement basic components of system software, including the operating system kernel and language compilers, on the prototype hardware systems
- Implement several important scientific applications on the resulting systems, using available language paradigms and the best available algorithms
- Make detailed measurements of application performance on the resulting system
- Identify principal hardware difficulties, operating system shortcomings, performance bottlenecks and reliability problems

Thrust 3: Technology into real applications Focus Areas

3.1 HECC Applications

3.2 Computational Science

3.3 Algorithm Improvement

3.1 HECC Applications

Programmatic/Technical Objectives: Identify key application drivers for future high-end systems, and demonstrate prototype implementations on available hardware platforms.

- Grand Challenge teams (DOE, NASA, NSF, NOAA, EPA)
- Scientific visualization and data management
- Cross-cutting technologies
- Problem solving environment for multi-discipline environmental modeling

3.2 Computational Science

Programmatic/Technical Objectives: Develop the practice of high-performance computational science as a third methodology of science, along with theory and experiment. Integrate computational science to effectively complement, and be complemented by, theory and experiment among the various scientific disciplines.

- Simulation Validation
- Simulation Exploitation

3.3 Algorithm Improvement

Programmatic/Technical Objectives: Determine key algorithms which potentially will or will not scale to PetaFLOPS computing and the level of effort involved. Explore concurrency, data locality, latency and synchronization, numerical accuracy, unstructured grid methods, and fault tolerance. Pursue detailed performance analyses, improvement metrics, numerical library routines, and new approaches, languages, and constructs, with an expanded research community.

- Determine scalability of key algorithms
- Investigate variants of known algorithms
- Assess novel algorithm potentials

Thrust 4: Infrastructure for research in HECC Focus Areas

4.1 Research Facilities

4.2 Access to large-scale test systems

4.3 Build on large-scale high performance networks

4.1 Research Facilities

Programmatic/Technical Objectives: To facilitate advancement in general research computational science by providing access to state-of-the-art facilities for scientists nationwide

- Procure and operate testbed system in facilities devoted to basic research in a variety of disciplines, such as NSF-PACI, DOE-NERSC, NASA-HPCC

4.2 Access to large-scale test systems

Programmatic/Technical Objectives: Facilitate access to state-of-the-art testbeds, as well as improved communication of scientific data and results between scientific teams, by utilizing the most advanced long-haul networking technology available

- Determine current infrastructure baseline, and constituent elements, with data sources; define methods of transforming raw data into indicators and metrics
- Determine annually the change in research infrastructure by deciding which expenditures and resources contribute to improving the infrastructure
- Compare and contrast U.S. HEC research infrastructure and its annual change relative to that of other nations
- Capture the lessons from current investments in HEC research infrastructure; package the lessons for dissemination across the U.S. HEC community
- Characterize the benefits derived from the research infrastructure

4.3 Build on large-scale high performance networks

Programmatic/Technical Objectives: Coordinate with other federal initiatives and interested external parties, providing an information exchange to facilitate the development and maintenance of an interconnection capacity that is sufficient to exploit the capabilities of large-scale computing.

- NGI coordination/interconnection

National Security Applications

- Nuclear weapons stockpile stewardship: Weapon effects simulation to extend stockpile in era of no new design
- Wide area imagery: Near real time analysis of imagery with 3D Resolution, 100,000 sq. mile coverage and high resolution
- Cryptology: Rapid decryption of multiple messages
- Vehicle and weapons design and test
  - High performance aircraft design and test: Full 3D multi-disciplinary (aero, structures, magnetics, propulsion, controls) simulation
  - Weapons systems such as high power RF weapons: End-to-end simulation to predict complex systems response to weapons effects
  - Target discrimination: Combat identification, signature extraction and rapid target insertion for advanced platforms

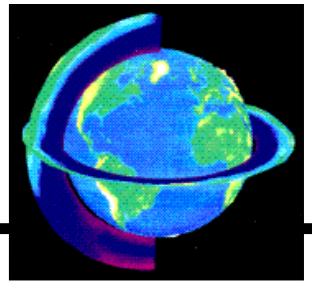
National Security Applications (con't)

- Intelligence data and information extraction
  - Remote sensing exploitation: Parameter analysis of chemical species and spectral band)
  - Chemical detection (FTIR remote sensor): Infrared image generation for radiometric, thermal and emissivity analyses
  - Intelligence data and Information extraction: Reduce time for analysis of complex sensor data from two weeks to real time
- Synthetic theater of war C 3 : Mission rehearsal and decision support for Desert Storm size scenarios in faster than real time

HECC Accomplishments
Specific Examples

- Prediction of Weather, Climate and Global Change
  - Accurate Ocean Models (10-by-10 mile resolution)
  - Accurate Hurricane Predictions (accurate 24- and 72- hour forecasts)
- Ecosystem Modeling, Environmental Monitoring
  - Accurate 3-D groundwater flow and transport models
  - Validated ozone concentration prediction in several major US cities
  - Simultaneous prediction of ozone and fine particles for assessment of human exposure and risk
- Product Design and Manufacturing
  - Environment for aircraft engine simulation reduces design time by 50%
- Biomedical Imaging, Molecular Biology and Health Care
  - Visible Man and Woman data sets digitized at sub-millimeter resolution
  - Estimation of 3-D molecular structure from sequence data support study of viral infections

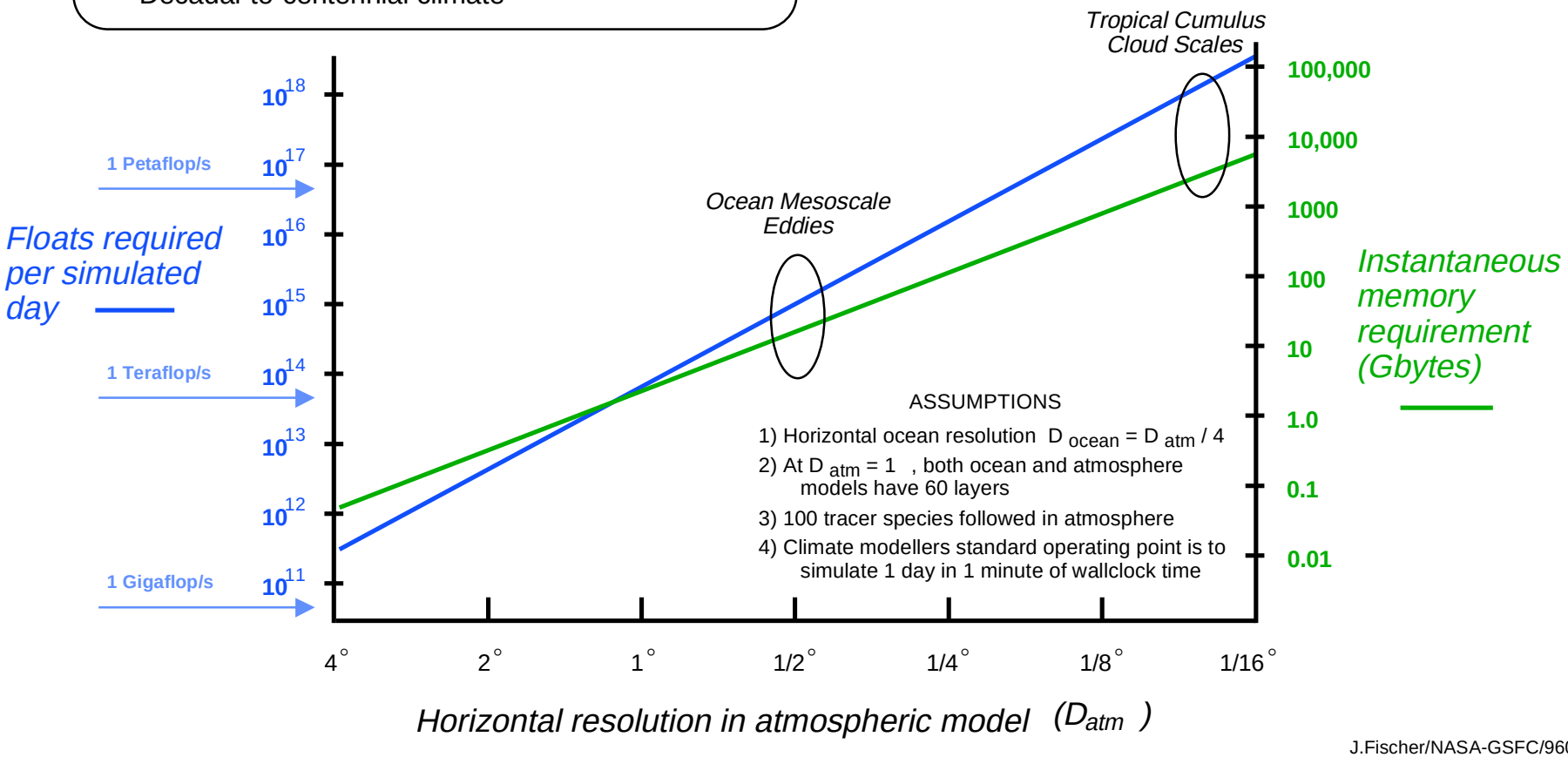
Computing Requirements for Coupled Global Circulation Models



U.S.Global Change Research Program

Areas of Great Scientific Challenge and Practical Significance

- Seasonal-to-interannual climate prediction
- Atmospheric chemistry
- Terrestrial and marine ecosystems
- Decadal-to-centennial climate



J.Fischer/NASA-GSFC/960910
P.MacNeice/HSTX
R.Mechoso/UCLA

NSF Grand Challenges

Astrophysics

- Turbulent compressible convection in stars
- Simulation of the formation of galaxies
- Interaction of two black holes

Structural biology

Remote electron microscopy
Biopolymer assemblies, e.g. protein aggregates
Enzyme simulation using molecular dynamics

Computational chemistry

Shell model for nuclear structure
Combustion, atmospheric chemistry
Collisions between electrons and molecules

Physiology

Simulation of joints such as the knee
Simulation of the cerebellar cortex

Aerodynamics

- Aeroelastic simulation of a full aircraft
- Direct numerical simulation of turbulence
- Electromagnetism

Ecosystems

Earthquake induced ground motion
Severe storm prediction
Simulation of landscape ecosystems

Emerging Computational Opportunities: (Requirements for Solution in 1 Week)