Computational and Data Grids

Distributed High-Performance Computing and Large-Scale Data Management for Science and Engineering

William E. Johnston, Dennis Gannon**, and Bill Nitzberg****

*National Energy Research Scientific Computing Division, Lawrence Berkeley National Laboratory and Numerical Aerospace Simulation Division, NASA Ames Research Center – wejohnston@lbl.gov

**University of Indiana and NAS, NASA Ames – gannon@cs.indiana.edu.

***MRJ Technology Solutions (NASA contract NAS2-14303) and NAS NASA, Ames – nitzberg@nas.nasa.gov

Outline

Overall Motivation and Goals

Large-scale science and engineering is done through the interaction of people, computing resources, information systems, and instruments, all of which geographically and organizationally dispersed.

The overall motivation for “Grids” is to facilitate the routine interactions of these resources to facilitate large-scale science and engineering.

Requirements

Analysis of several science and engineering scenarios provide various requirements for Grids. Further, these scenarios –

- **engineering design and multi-disciplinary science**
- **scientific data analysis and computational modeling**
- **real-time data analysis (e.g. on-line instruments)**
- **coupling experiments and computational models**
- **generation and management of large, complex data archives**

share certain characteristics. For example:

- **Both multi-disciplinary science and engineering design involves *multiple datasets* – e.g. geometry (structure) and performance – that are maintained by discipline experts at different sites, and must be *accessed and updated by collaborating analysts*.**
- ***Complex workflow scenarios* involving many compute and data intensive steps must be managed – e.g. in mutli-disciplinary simulations and laboratory and data analysis protocols.**

Requirements (cont.)

- **Existing heterogeneous *sub-component simulations need to be coupled and operated simultaneously* in order to provide whole system simulations (e.g. "multi-disciplinary optimization").**

- **Interfaces to computational and data tools must provide *appropriate levels of abstraction* for discipline problem solvers.**

- **Techniques are needed to *search, interpret, and fuse multiple remote data archives*.**

- **Scientists and engineers must be able to *securely share* all aspects of their work process.**

Requirements (cont.)

- ***Data streams from instrument systems*** **must be available in real-time to computational data analysis systems, and also via well catalogued databases.**
- **Tools and services for *fault management* and recovery are required for both applications and infrastructure**

These general requirements imply certain capabilities that must be provided by Grids in order to support the interactions of the instruments, people, information systems, and computing resources needed to facilitate distributed science and engineering.

There is an additional set of requirements from the tool developers – the application domain computational scientists – primarily in the area of supporting code development and execution environment access and management. These will not be discussed here, but are also major drivers for developing Grid functionality.

What are "Grids"?

Science Grids are distributed, high performance computing and data handling infrastructure that

- **is persistent and supported**
- **incorporates geographically and organizationally dispersed, heterogeneous resources:**
  - **computing systems**
  - **storage systems**
  - **instruments and other real-time data sources**
  - **human collaborators**
  - **communications systems**
- **provides common interfaces for all resources**

Grid components and infrastructure includes:

- Tools for constructing collaborative, application oriented Problem Solving Environments / Frameworks
- Programming environments, tools, and services providing various approaches for building applications that use distributed computing systems, federated data sources, and other distributed resources

What are "Grids"? (cont.)

- **Grid Common Services / runtime environment:**
  A comprehensive and consistent set of location independent tools and services for accessing and managing dynamic collections of widely distributed resources in order to support
  - **resource discovery**
  - **co-scheduling**
  - **job management**
  - **uniform access to tertiary storage**
  - **runtime access and monitoring**
  - **access control**

What are "Grids"? (cont.)

- **Operational services including**
  - **system management tools for distributed systems and distributed resources**
  - **user support**
  - **accounting and auditing**
  - **security, including support for**
    - **+strong and location independent user authentication and authorization**
    - **+overall system security services**

Vision for Science Grids

Computing and data Grids in the service of science will provide significant new capabilities to scientists and engineers by facilitating *routine construction of information based problem solving environments / frameworks that knit together widely distributed computing, data, and instrument systems*

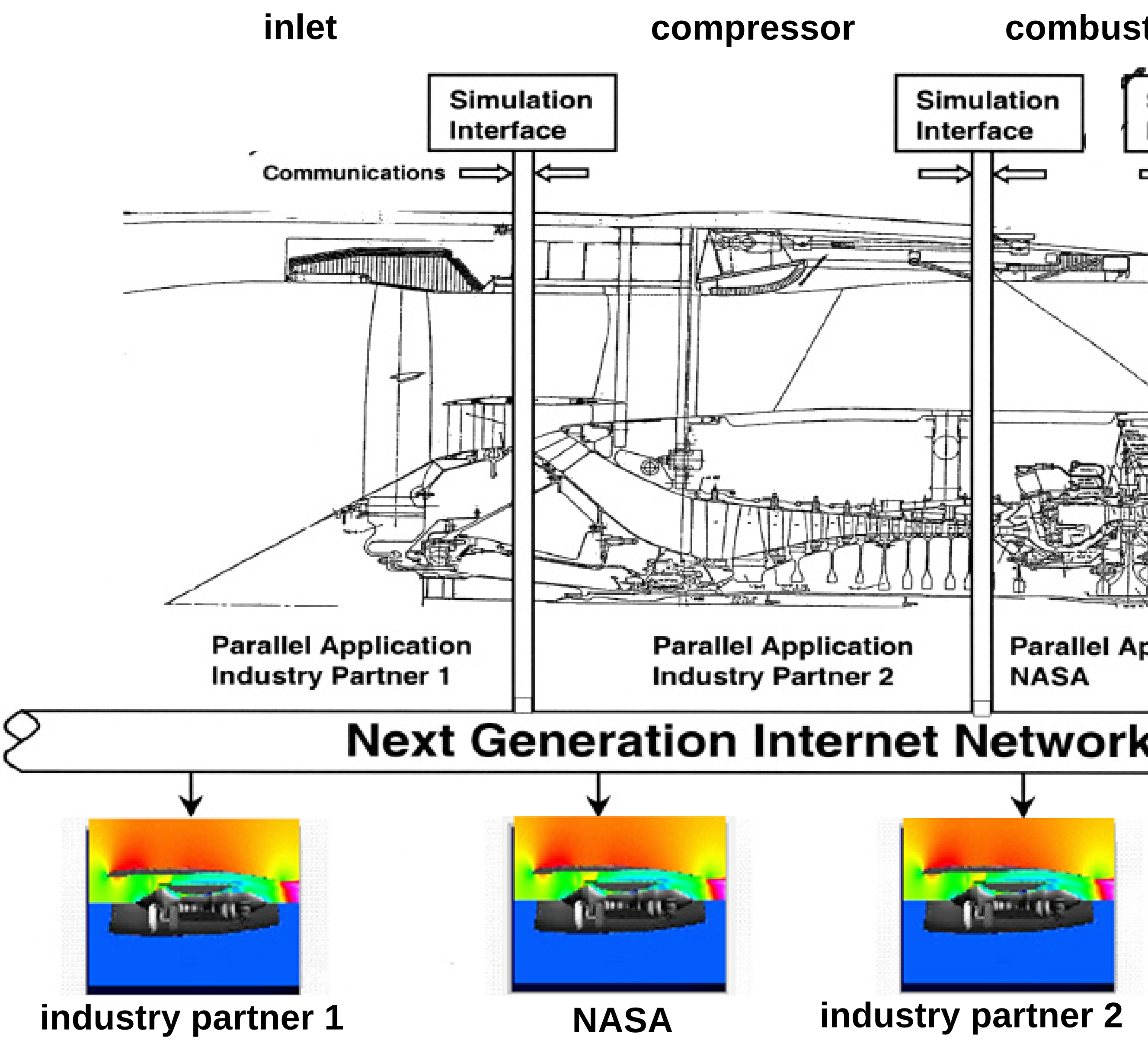
– esp. supercomputers, petabyte storage systems, and unique national-scale instruments –

together with human resources, *into aggregated systems that can address complex and large-scale computing and data analysis problems* beyond what is possible today.

Examples of Various Sorts of "Scale" in Grids

- **Large computing capacity through the aggregation of many resource to support large-scale computational problems.**
  - **E.g.: Coupled, multidisciplinary simulations too large for single systems**
    - **- multi-component turbomachine simulation (NPSS at GRC)**
    - **- multiple model climate simulations**

Collaborative Multi-Component Distributed Simulation



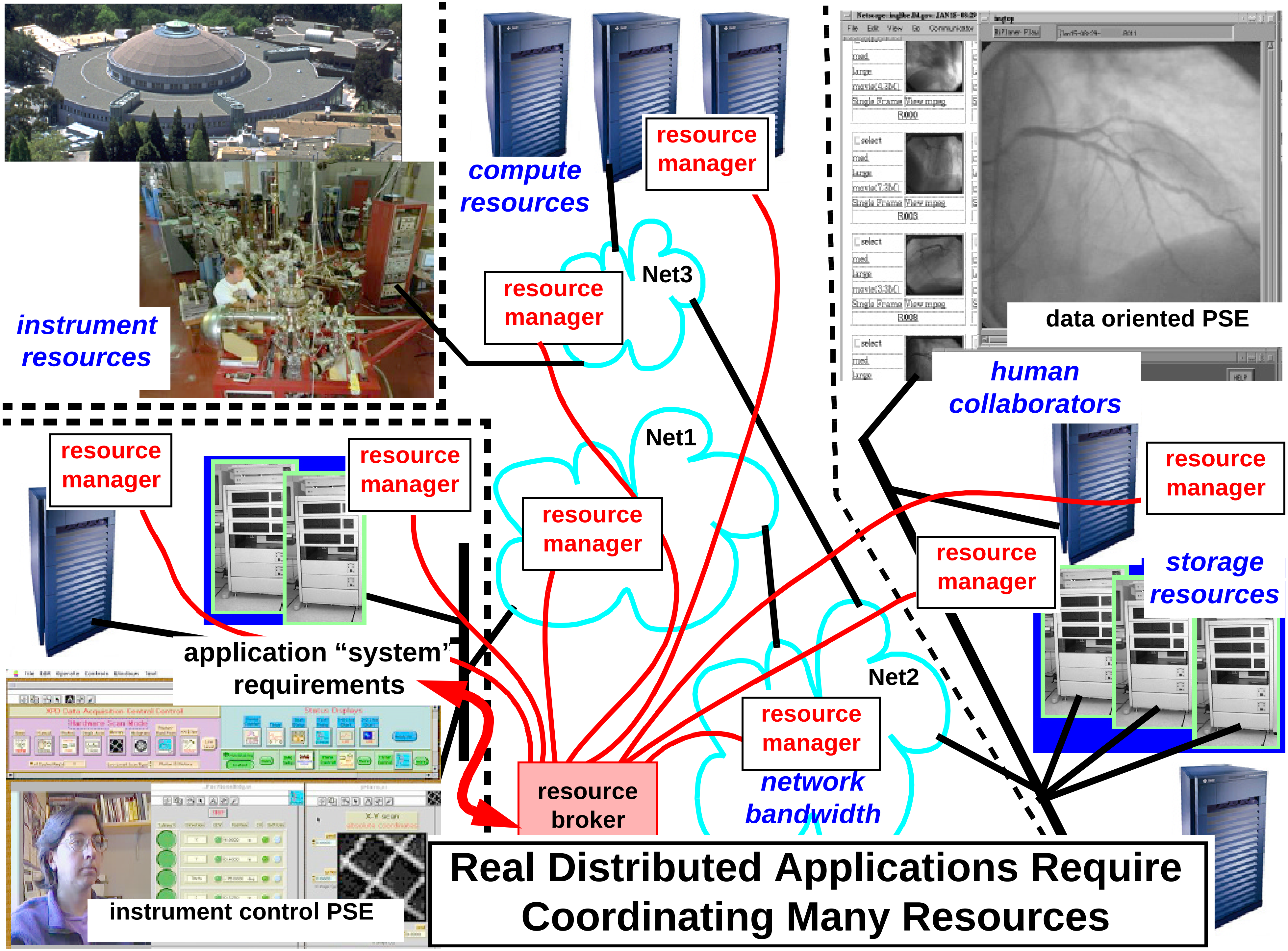
full simulation viewed at all collaborating sites

NPSS
Computing and Interdisciplinary Systems Office
NASA Glenn Research Center

(A multi-disciplinary simulation: the four regimes of an operating jet engine.)

Scale in Grids (cont.)

- **Coupling large-scale computing and data systems to scientific and engineering instruments requires that many heterogeneous resources be located, co-scheduled, and managed, including collaborators, instruments, bandwidth, and storage and computational systems**
  - **E.g.: real-time interaction with experiments through real-time data analysis and interpretation presented to the experimentalist in ways that allow direct interaction with the experiment (instead of just with the instrument controls)**
  - **E.g.: real-time processing and distribution of satellite data feeds**



Real Distributed Applications Require Coordinating Many Resources

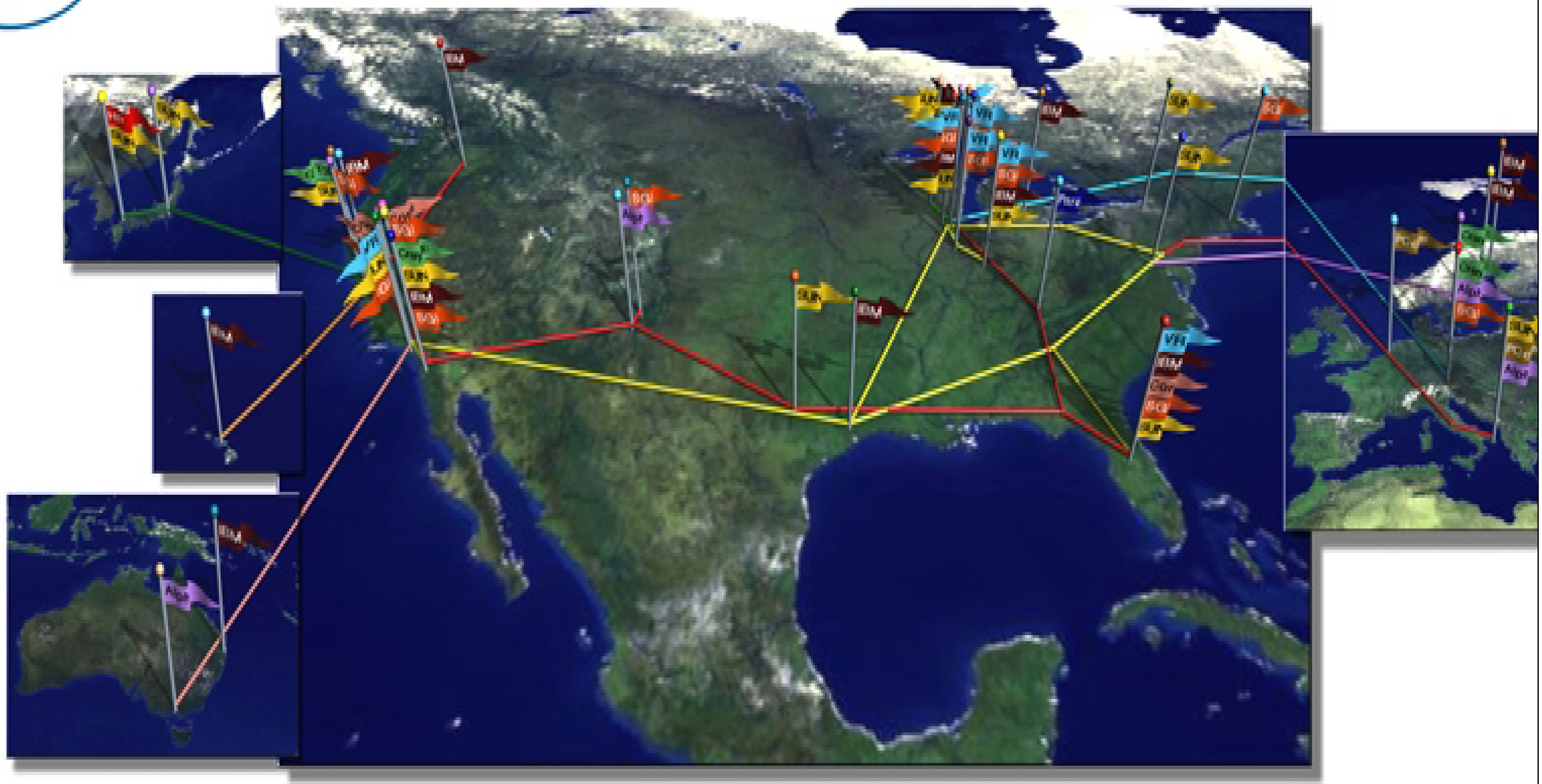
Scale in Grids (cont.)

- **Geographic and organizational scope**
  - **Grids are built from, and designed to manage, organizationally and geographically dispersed components**

the globus project
www.globus.org

GUSTO Computational Grid Testbed

as of November 1998



Geographic and Organizational Complexity is the Norm in Grids.

Scale in Grids (cont.)

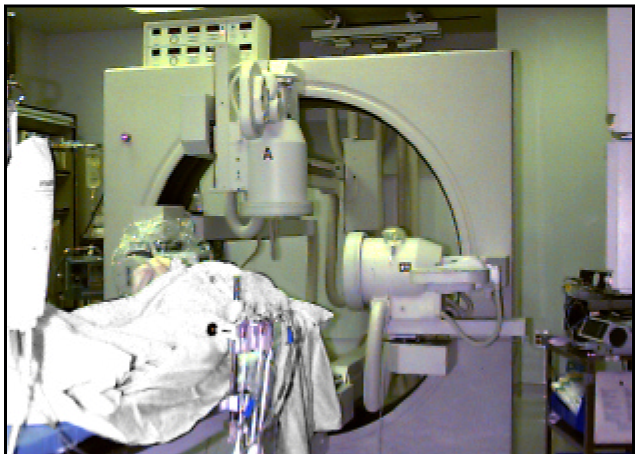
¨ **Data intensive computing: use of widely distributed, federated data archives**

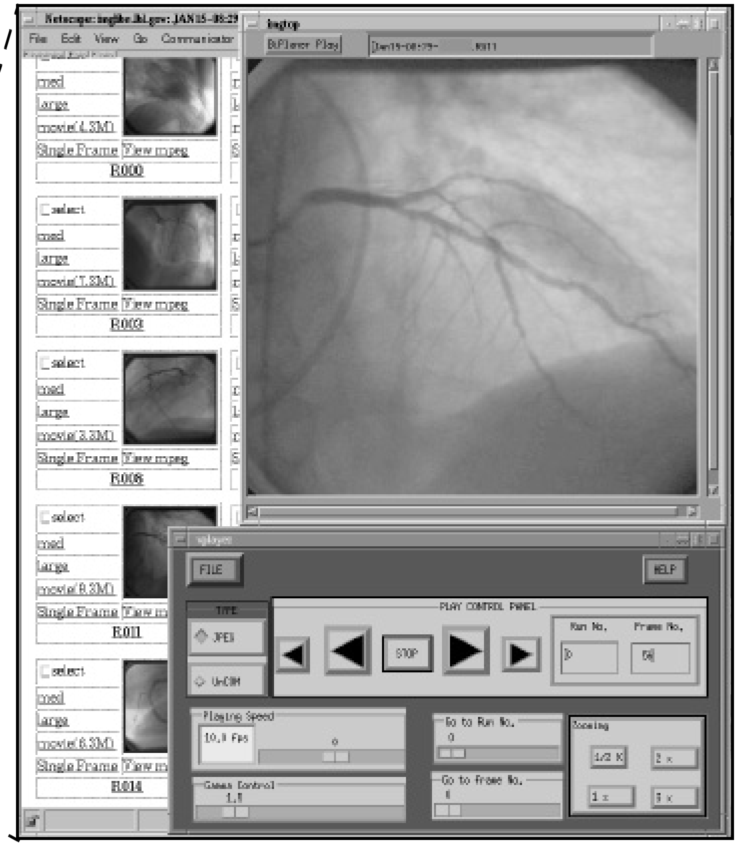
- **Example: Real-time digital libraries for on-line, high data-rate instruments [7]**
  - **on-line, real-time, high data-rate instrument**
  - **management of large data sets in wide area**
  - **optical WDM metropolitan area network (now part of NGI)**
  - **remote data analysis followed by automatic data cataloguing and archiving**
  - **remote data users**
  - **widely distributed, high performance “application-level” cache**
  - **a “data flow” architecture for high data-rate on-line instrumentation systems**

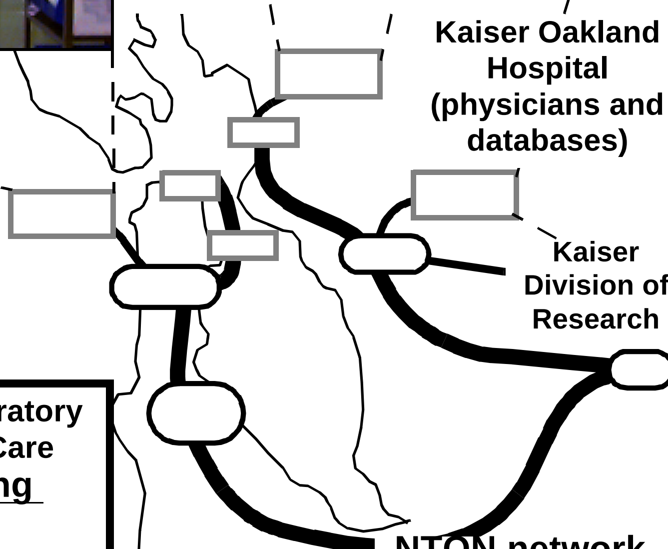
WALDO real-time digital library system and DPSS distributed cache [7] for data cataloguing and storage

Compute servers for data analysis and transformation

Kaiser San Francisco Hospital Cardiac Catheterization Lab (X-ray video imaging system, ≈ 130 mbit/s, 50% duty cycle 8-10 hr/day)





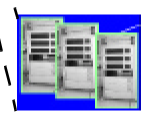


Kaiser Oakland Hospital (physicians and databases)

Kaiser Division of Research

NTON network testbed

The PSE: Automatically generated user interfaces providing indexed access to the large data objects (the X-ray video) and to various derived data.

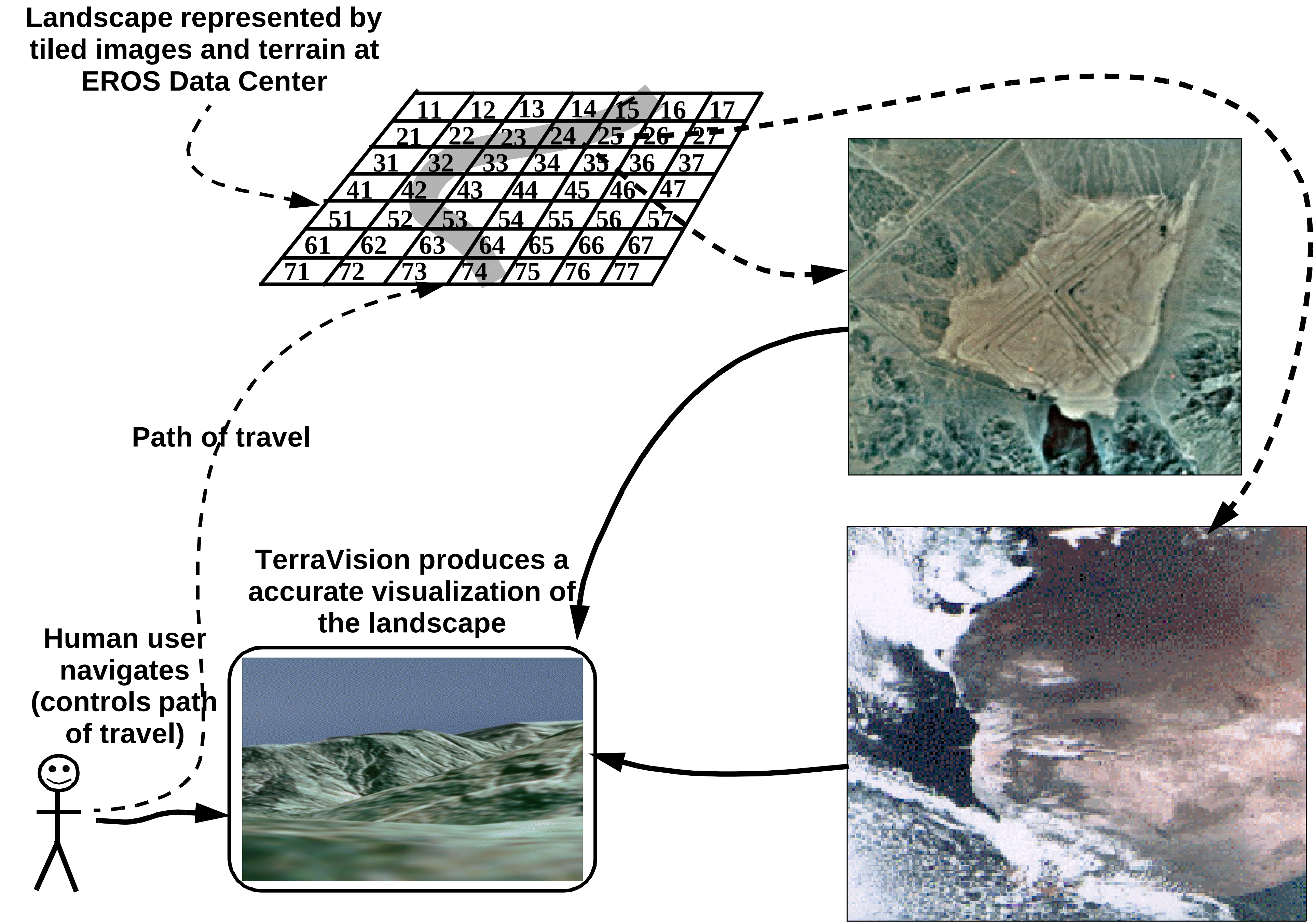


Lawrence Berkeley National Laboratory and Kaiser Permanente Health Care

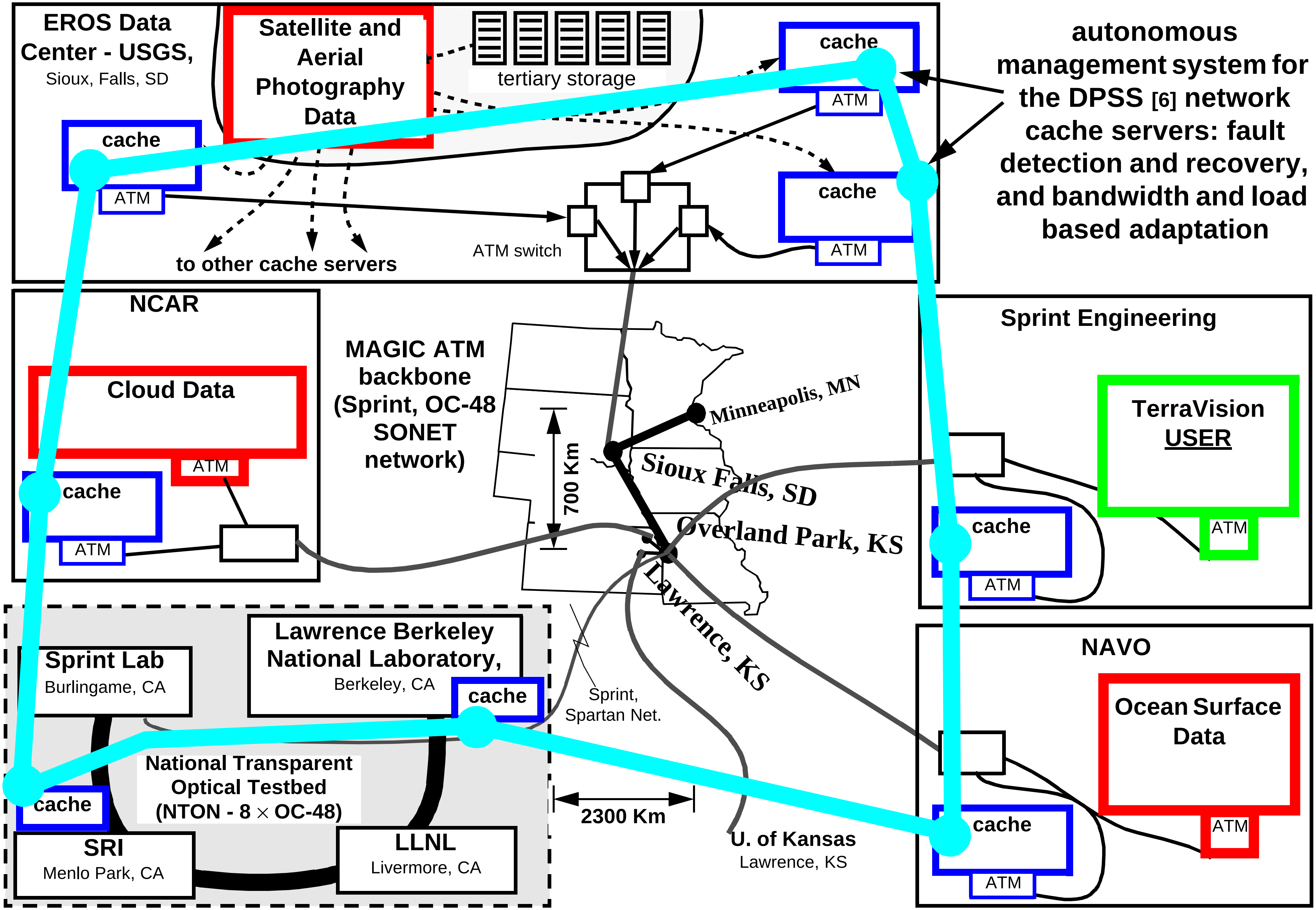
<u>On-line Health Care Imaging Experiment in the San Francisco Bay Area</u>

Scale in Grids - Data Intensive Computing (cont.)

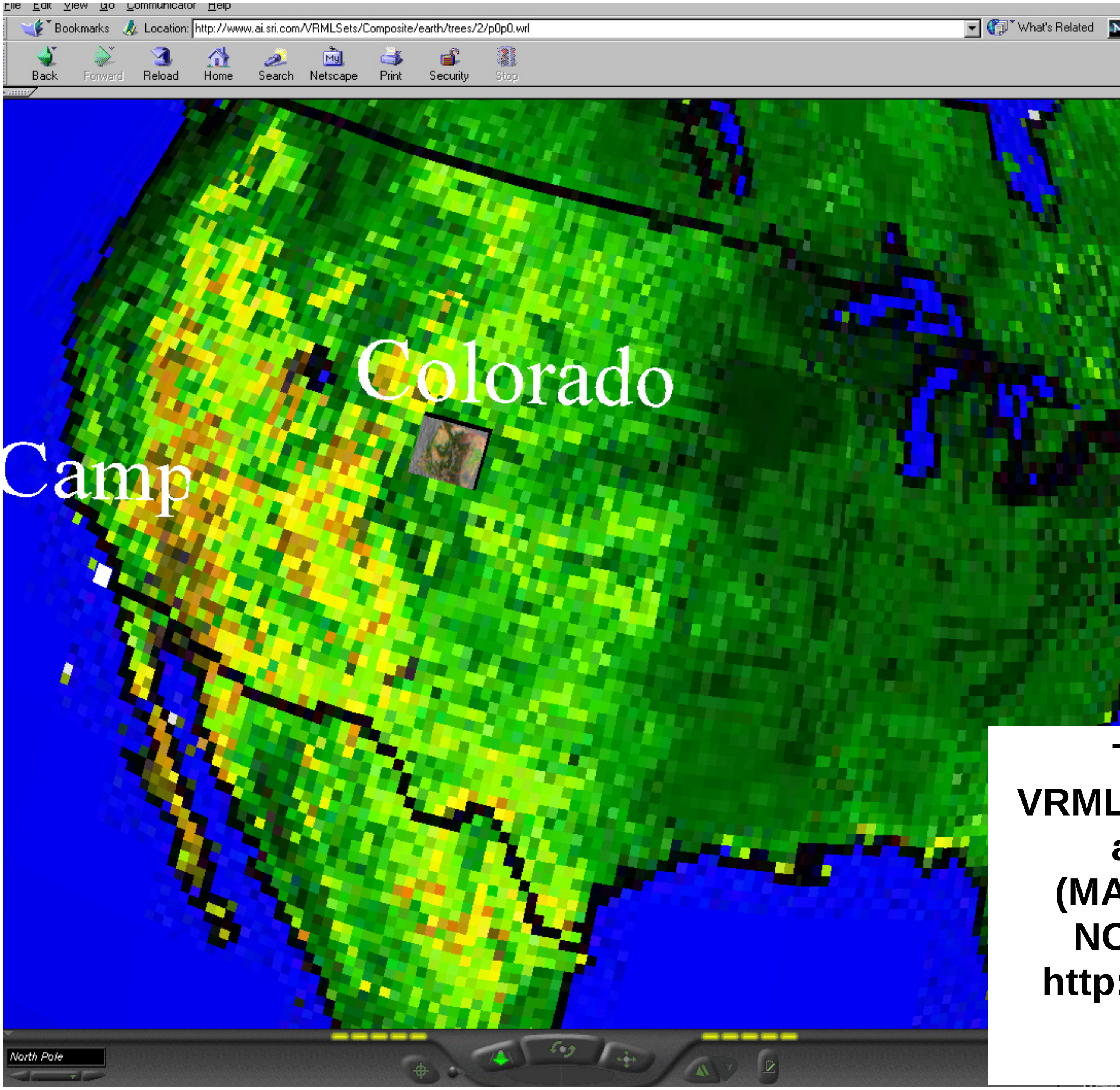
- **Example: High data-rate distributed data management and federated access for archived satellite and aerial imagery, digital terrain data, and atmospheric data ([8] and [9])**
  - **on-line, real-time access to multiple environmental data sets that are (and always will be) maintained by domain experts at their own sites.**
  - **on demand, real-time interactive exploration of an operational environment supporting, e.g., military operations and community emergency services**
  - **aggregation of multiple, widely distributed, multi-discipline data sets**
  - **DARPA MAGIC testbed consortium (see www.magic.net) developed distributed services, data and visualization from EROS Data Center, NCAR, NAVO, SRI (collab. with NASA NREN)**
  - **MAGIC wide-area, gigabit network testbed is now part of NGI**



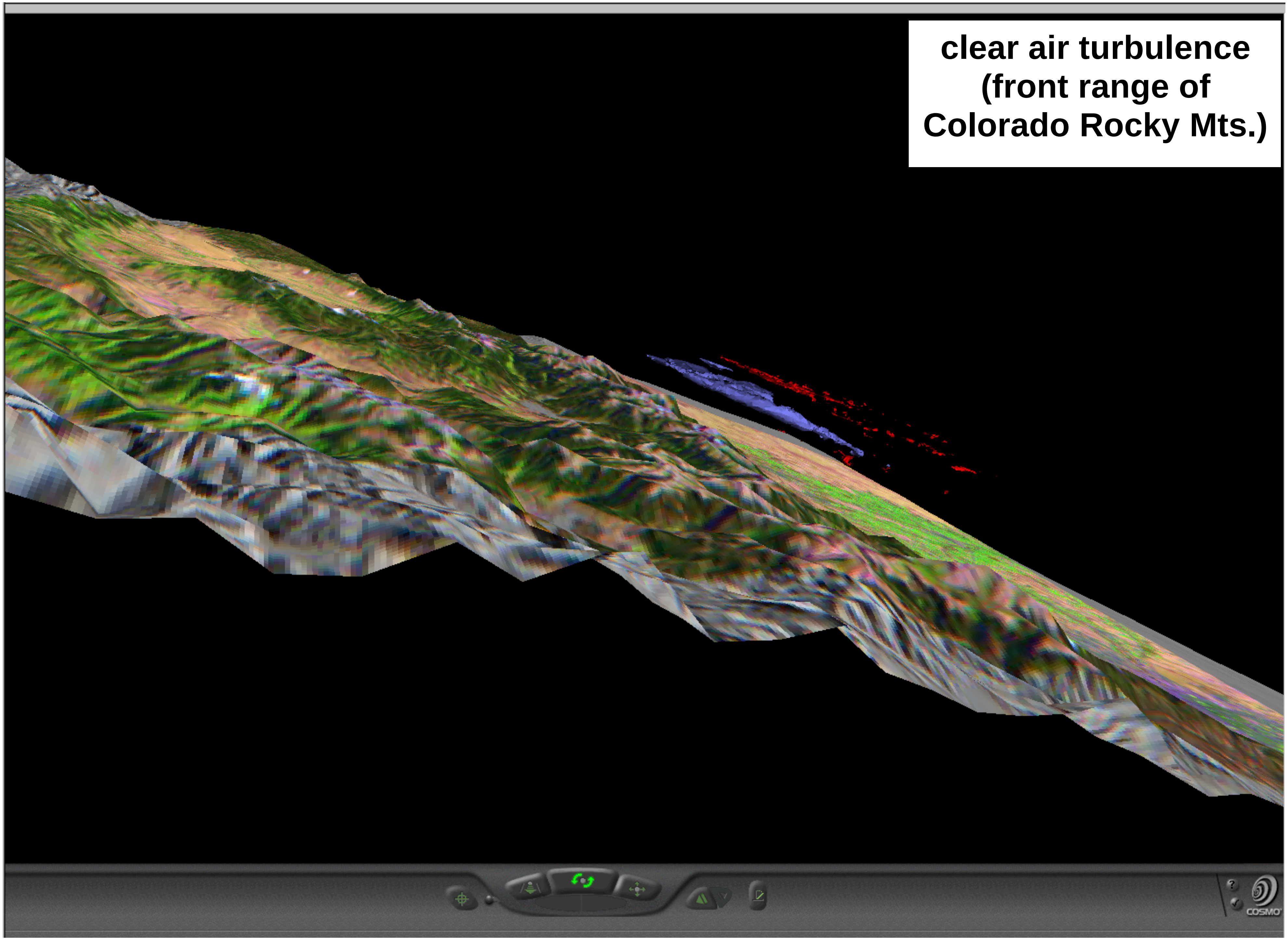
TerraVision Provides Real-time Visualization of Aggregated Data



The MAGIC Testbed Distributed Application Environment

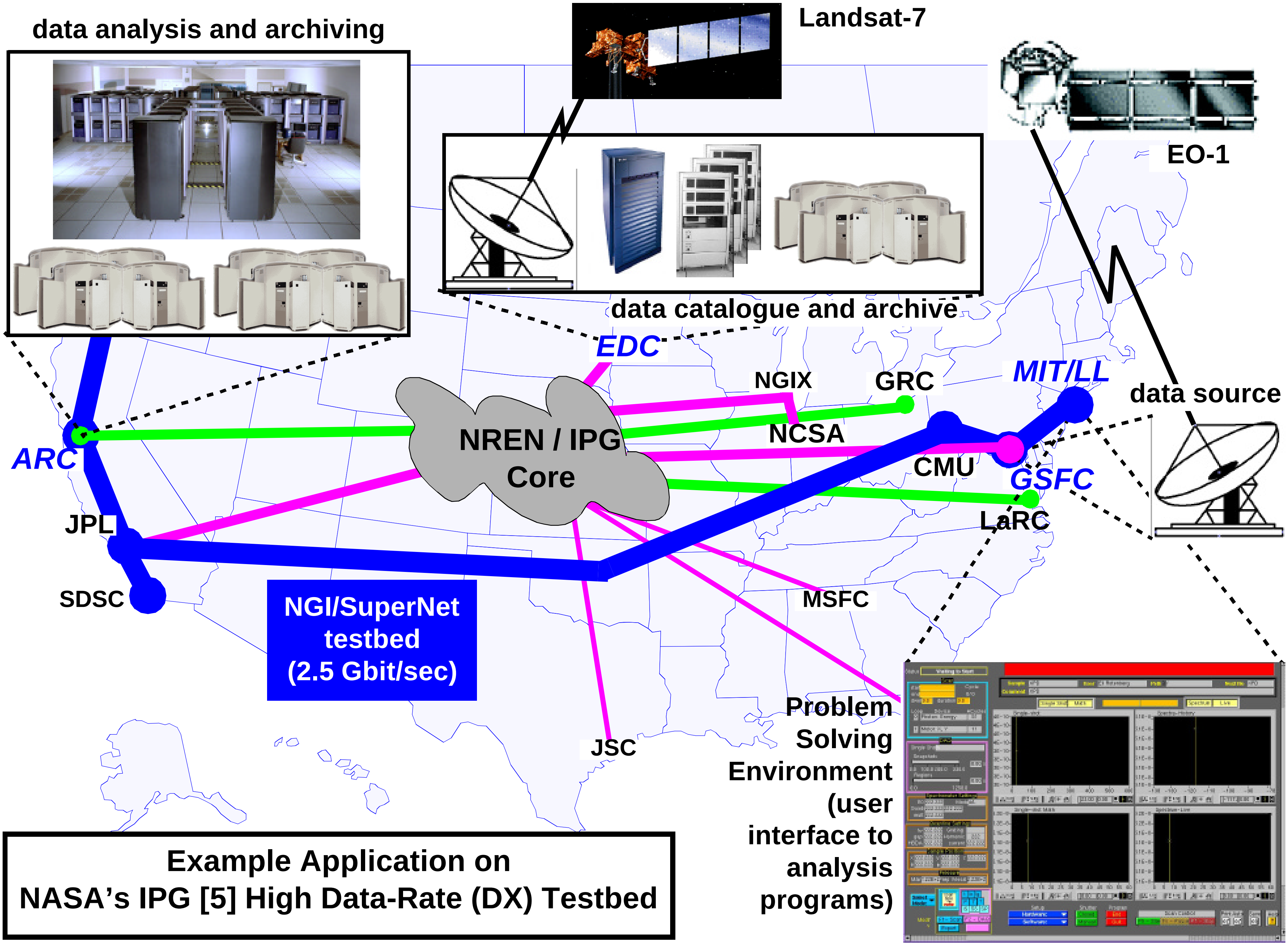


TerraVision-2:
VRML based data fusion and browsing.
(MAGIC consortium, NCAR, and NAVO: http://www.ai.sri.com/ TerraVision/)


clear air turbulence
(front range of
Colorado Rocky Mts.)

Scale in Grids (cont.)

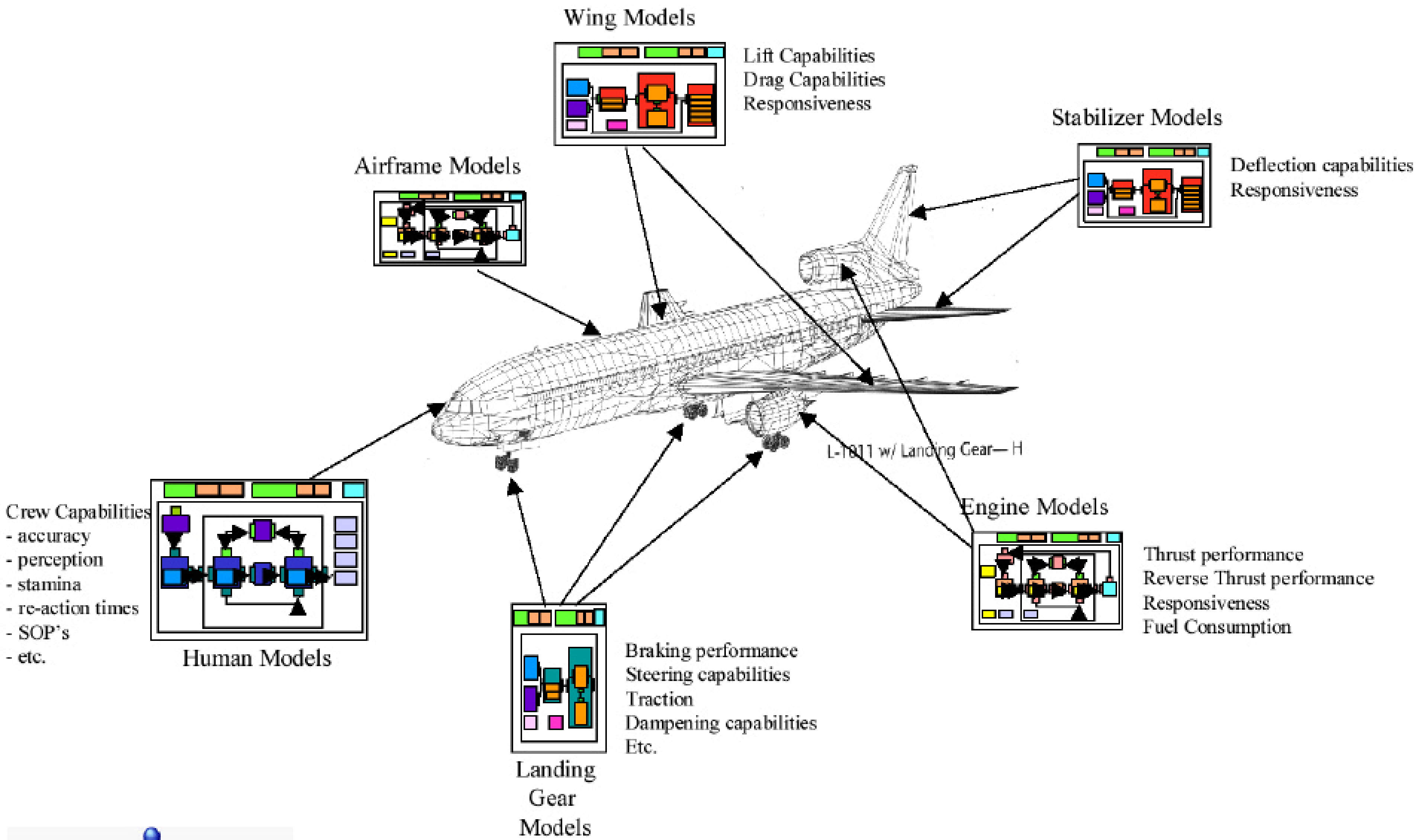
- **Large-scale Grids will be needed for many real problems that involve geographic and organizational diversity *plus* compute and data intensive applications**
  - **E.g.: satellite data processing system:**
    - **simultaneous, on-line analysis of several satellite sensor systems**
    - **multiple satellites feeding data into the NASA Grid which uses the DARPA NGI/SuperNet testbed for high data-rate systems**
    - **data is analyzed, catalogued, and archived at multiple locations, depending on which sites have the curating expertise for the various data types**


data analysis and archiving
Landsat-7
EO-1
data catalogue and archive
EDC
NGIX
GRC
MIT/LL
data source
NCSA
NREN / IPG
Core
CMU
GSFC
LaRC
ARC
JPL
SDSC
NGI/SuperNet
testbed
(2.5 Gbit/sec)
MSFC
JSC
Problem
Solving
Environment
(user
interface to
analysis
programs)
Example Application on
NASA's IPG [5] High Data-Rate (DX) Testbed

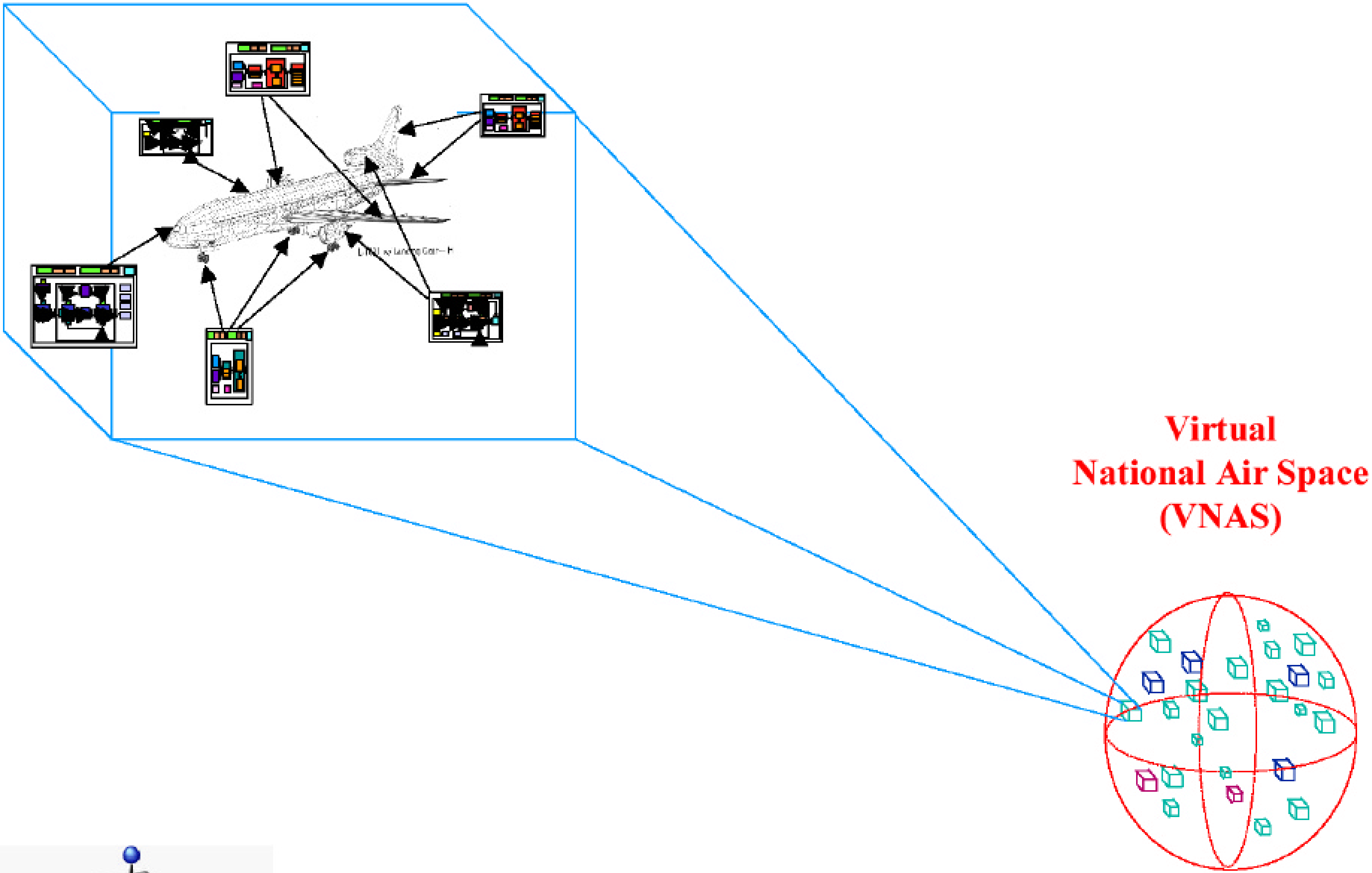
Large-Scale Grids (cont.)

- **E.g.: Virtual National Airspace Simulation**
  - **Yuri Gawdiak and Anthony Lisotta, NASA Ames**
  - **simultaneous access to metrological, topological, aircraft performance, and flight path scheduling databases supporting a National Air Space Simulation system**

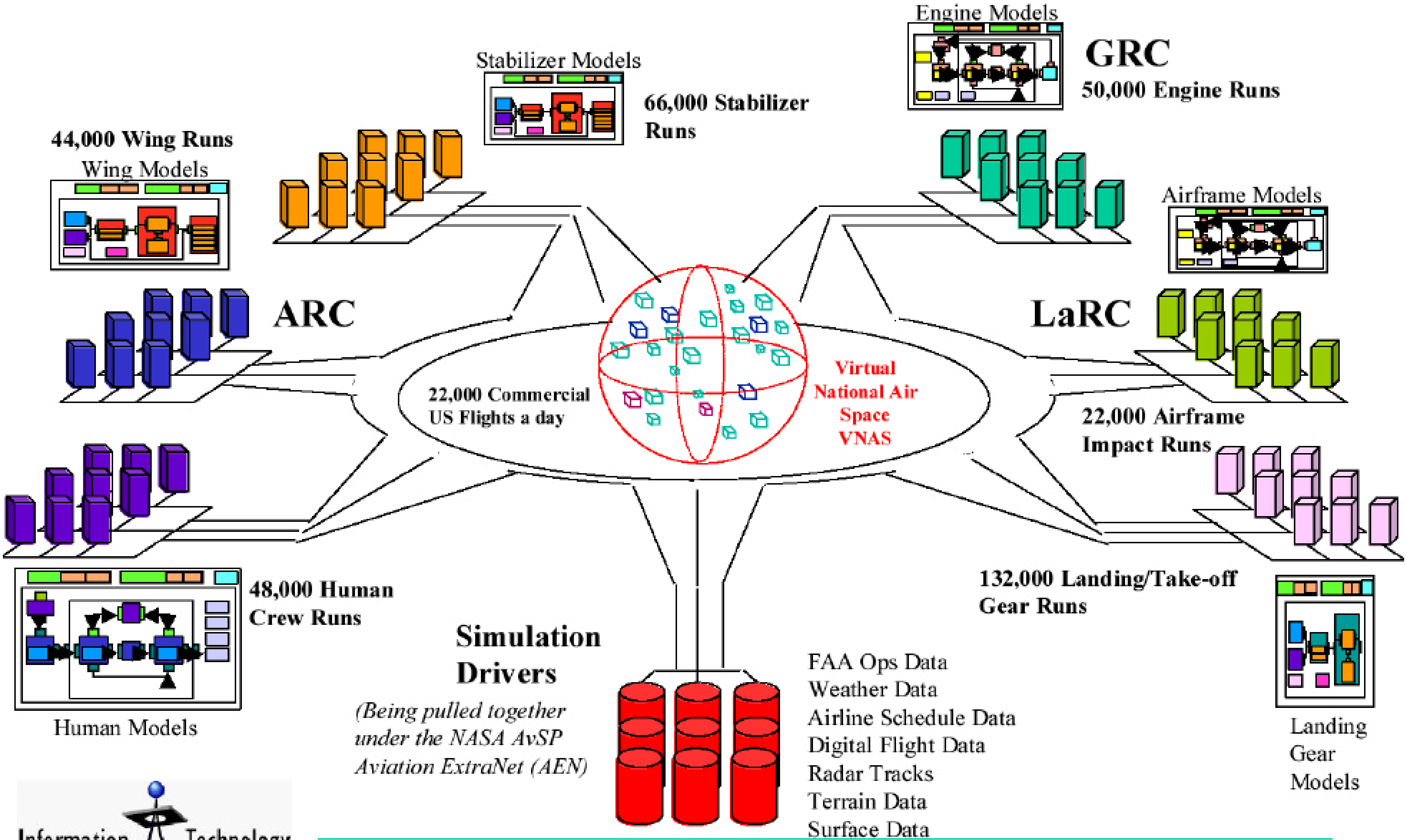
Performance Characteristics for Fast-Time Simulations



National Air Space (NAS) Simulation Environment



Daily VNAS Simulation Baseline Generation



Scale in Grids (cont.)

- **Highly interactive, augmented reality and virtual reality based remote collaborations**
  - **E.g.: Ames / Boeing Remote Help Desk providing field maintenance use of coupled video and NDI to a remote, on-line airframe structures expert who uses this data to index into detailed design databases, and returns 3D internal aircraft geometry to the field.**
  - **E.g.: Collaborative, distributed exploration of high energy physics events and detector operation**

Scale in Grids (cont.)

- **Single computational problems too large for any single system**
  - **E.g.: The rotocraft reference calculation – sufficiently fine geometry that solution no longer changes with cell size**

Grids also have the potential to *provide pools of resources that could be called on in extraordinary / rapid response situations* (such as disaster response) because they can provide:

- **common interfaces and access mechanisms**
- **standardized management**
- **uniform user authentication and authorization**

for large collections of distributed resources (whether or not they normally function in concert).

What Will Grids do for Us? (cont.)

Fully implemented Grid services will also facilitate automatic fault management and workflow management in extremely remote systems such as isolated environments and satellite based sub-Grids.

What Must Grids Provide?

To satisfy requirements arising from these types of applications, certain functionality must be provided. *Example* functionality and what it facilitates includes:

- **Modular toolkits for building PSE/Frameworks that provide workflow management, application code composition, access control, and collaboration**
  - ⇒ **the "user interface to the Grid"**
  - ⇒ **knowledge based workflow control (required for: fault management ("FM") and extremely remote systems ("ERS"))**
- **High throughput job managers**
  - ⇒ **specialized PSEs for parameter studies**

What Must Grids Provide (cont.)

- **Resource discovery and brokering, advance reservation, and co-scheduling for all resources**
  - ⇒ large-scale computing through aggregation
  - ⇒ on-demand and scheduled, dynamic system construction
  - ⇒ facilitates fault recovery (FM & ERS)
- **Monitoring for performance tuning, and fault detection and management**
  - ⇒ construction of reliable, production quality applications
  - ⇒ supports autonomous system management (FM & ERS)
- **End-to-end high bandwidth between resources**
  - ⇒ support for high data-rate distributed applications
  - ⇒ coupling of remote instruments with large-scale computing
  - ⇒ accommodation of very long RTT communication (ERS)

What Must Grids Provide (cont.)

- **Use of multi-source data resources**
  - ⇒ **federating datasets to support multi-disciplinary simulation**
- **Accommodation of "legacy" codes**
  - ⇒ **"wrapping" legacy codes to incorporate them into the compositional ("building block") mechanisms of PSEs**
  - ⇒ **assistance for porting code from the older small processor-count vector systems to newer high processor count shared memory and distributed memory architectures**
- **Data location management and optimized remote data access**
  - ⇒ **automatic location management to minimize data movement and data transfer time when CPUs and data archives are in different geographic locations (ERS)**

What Must Grids Provide (cont.)

- **Global event management**
  - ⇒ **synchronization of widely distributed processes and data sets to support consistency/repeatability of results, and use of “independent” data sources**
  - ⇒ **primary information source for workflow management (FM & ERS)**
- **Grid enabled / aware algorithms**
  - ⇒ **distributing single codes across Grids requires new techniques**

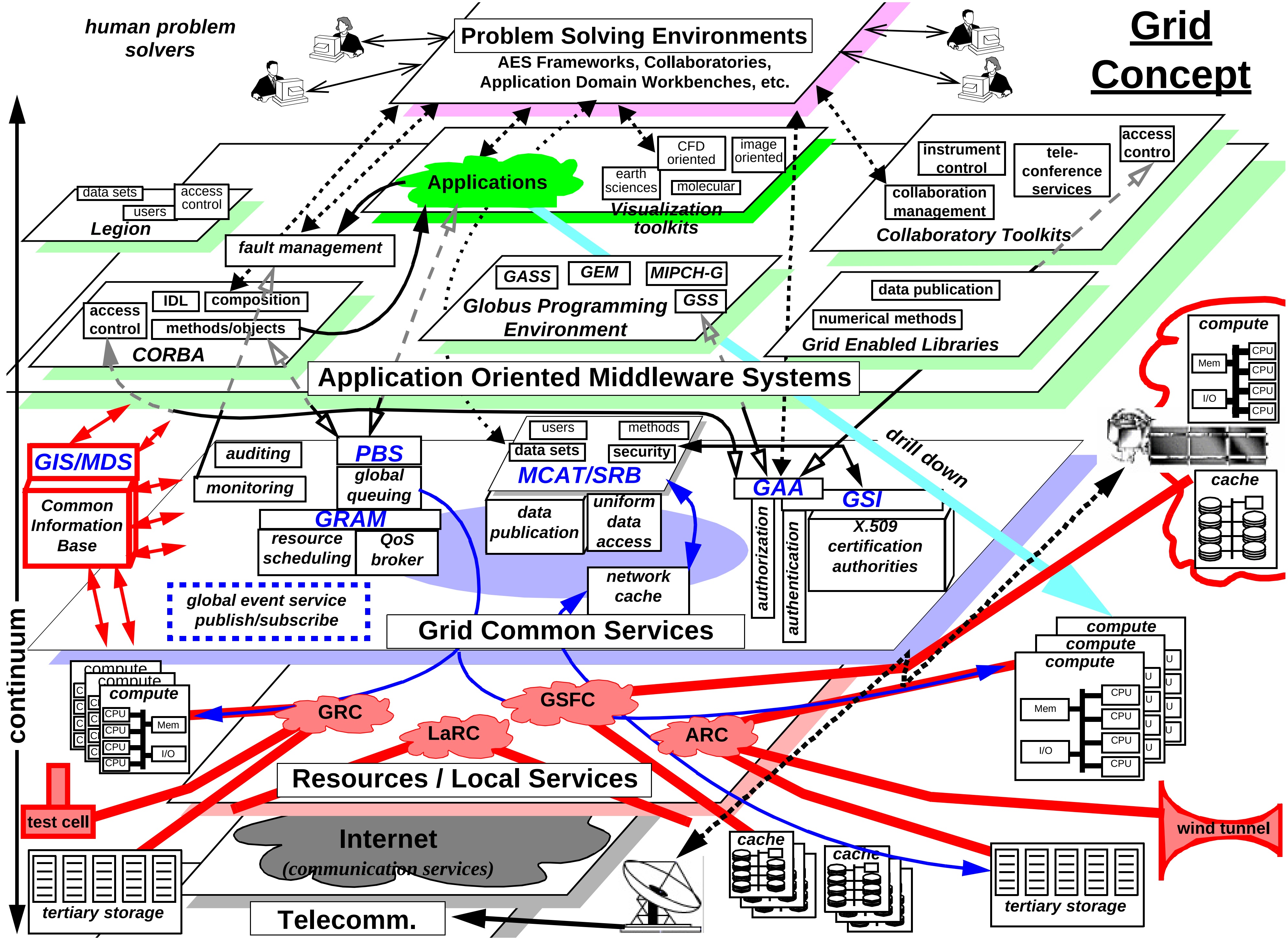
What Must Grids Provide (cont.)

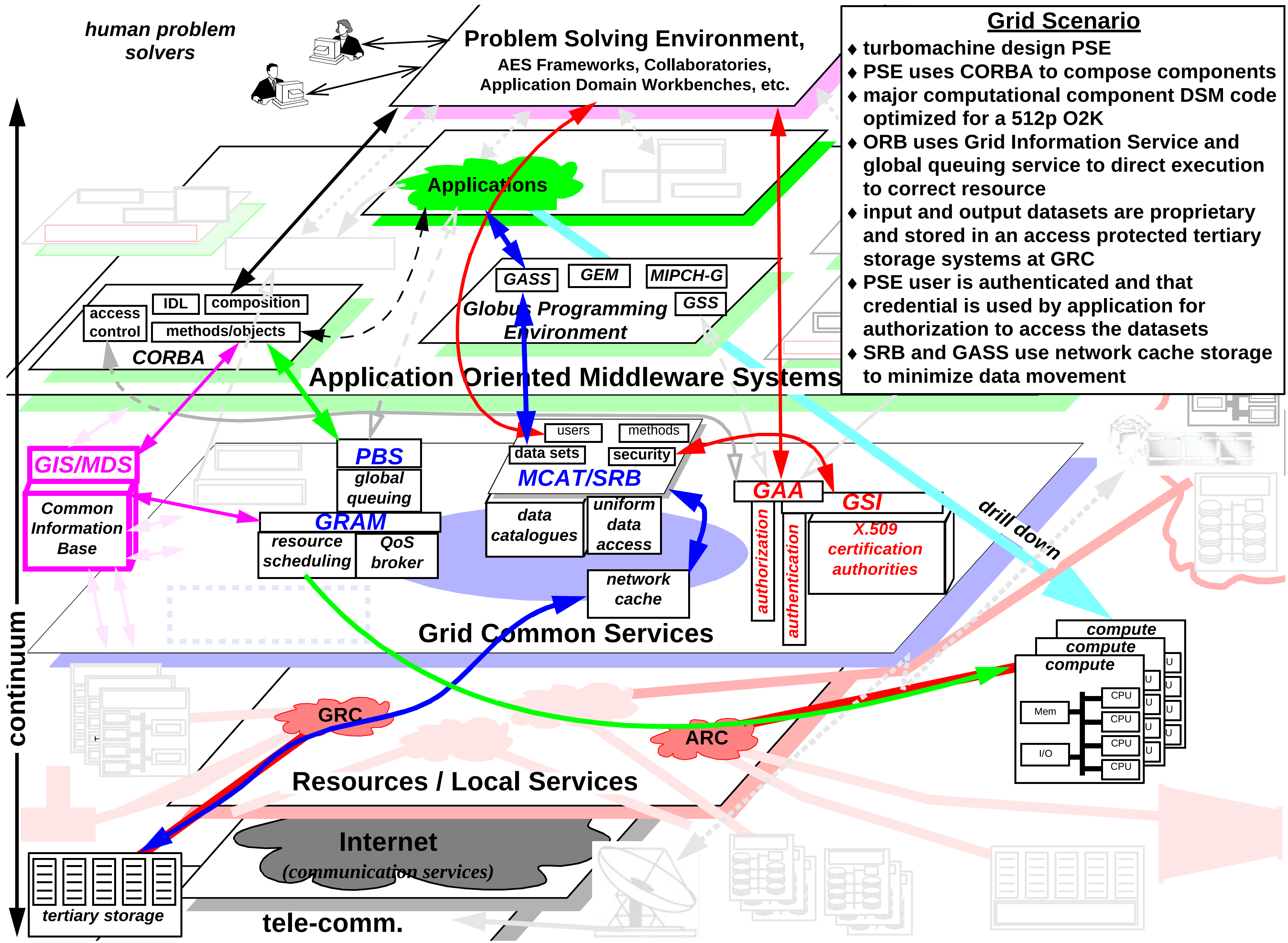
- **Security and infrastructure protection**
  - ⇒ **assurance of resource use and function in the semi-open environment of science and engineering R&D environments**
  - ⇒ **secure autonomous operation (ERS)**
- **Access control mechanisms**
  - ⇒ **management of access rights by data / resource stakeholders**
  - ⇒ **positive identification of users and management of user attributes pertaining to access rights**
- **Operational procedures and tools**
  - ⇒ **management of cross-organizational resources**
  - ⇒ **management of widely distributed resources**
- **User support, allocation management, and accounting**

The Components of a Grid

- *A conceptual framework for describing / organizing* **the functional components (see figure)**
- **Toolkits for building** ***PSEs / Frameworks***
- **Multiple** ***middleware*** **systems (code development)**
- **Grid enabled** ***toolkits / libraries***
- ***Grid Common Services*** **(mostly resource access and management related)**
- ***Resources*** **(compute, data, instruments, humans)**
- ***Operational infrastructure*** **(e.g., auditing, security, access control, user and system support)**
- ***Testbeds and prototypes***
- ***R&D*** **for new capabilities**

Grid Concept





Grid Scenario

- turbomachine design PSE
- PSE uses CORBA to compose components
- major computational component DSM code optimized for a 512p O2K
- ORB uses Grid Information Service and global queuing service to direct execution to correct resource
- input and output datasets are proprietary and stored in an access protected tertiary storage systems at GRC
- PSE user is authenticated and that credential is used by application for authorization to access the datasets
- SRB and GASS use network cache storage to minimize data movement

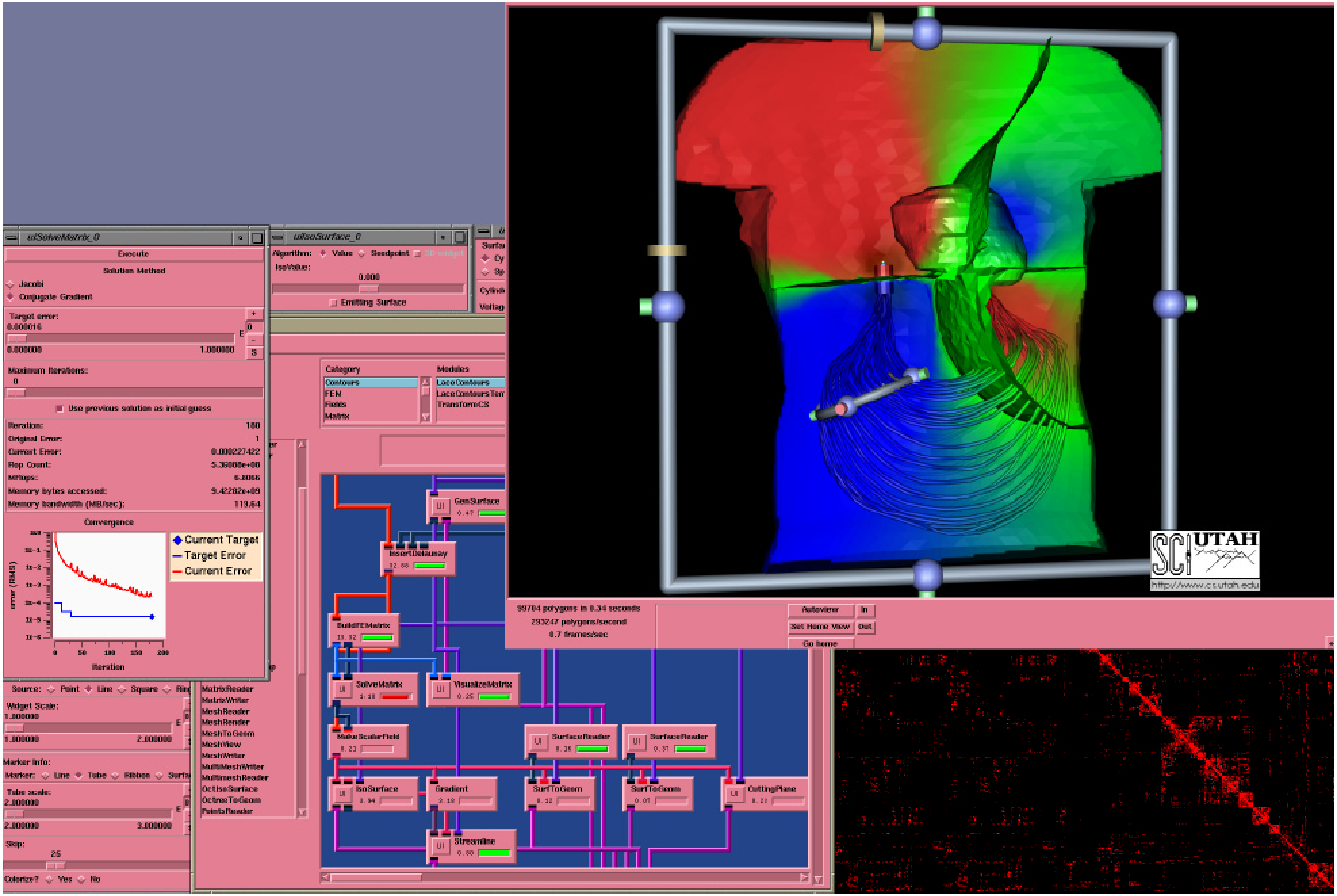
Problem Solving Environments / Application Oriented Frameworks

The PSE layer:

- **provides the scientist's / engineer's application domain-specific view of tools (e.g. simulations, databases, instruments) and Grid services and facilitates organizing and making tools and Grid services accessible in ways that correspond to the conceptual model of the problem being solved**
- **facilitates collaboration with others working on the same problem.**

Basic functionality includes, e.g., mechanisms for:

- **program and tool controls and interfaces**
- **composing code units**
- **workflow protocol definition and management**
- **access to Grid services such as data cataloguing and data management functions**
- **collaborative sharing that integrates with the other components of the PSE**



Example Problem Solving Environment. [27]

Applications

Domain-specific applications / "tools", e.g. numerical simulations and data analysis code that operate in Grid environments, are managed by the PSE / Framework. Applications, in turn, use Grid middleware services to facilitate their construction and operation.

Application Oriented Middleware Services

Middleware services provide different programming and service styles for application developers to access the basic Grid services.

- ♦ **By design, this layer supports heterogeneous approaches because application builders use different approaches to writing code.**
- ♦ **The PSE layer above this provides uniformity in that different applications and services must be able to be combined in the PSE to build problem solving systems.**
- ♦ **The Grid Common Services layer below this provides uniform access to resources**

Application Oriented Middleware Services (cont.)

These services include, e.g.:

- **Grid enabled programming environments for building new applications (e.g. Globus/MPI [2], CORBA, Legion [11], Java) as well as native MPI, DSM, etc.**
- **tools and techniques for accommodating (wrapping) "legacy" applications**
- **Grid enabled functions (e.g. visualization, remote instrument operation, numerical methods, data cataloguing services, remote instrument control services, debugging, etc.)**
- **compilation environment management**

Grid Common Services

Grid common services are independent, but consistent, services and infrastructure for supporting distributed resource access and management.

We divide these services into:

- **execution management**
- **runtime support**
- **environment management**

Grid Common Services: Execution Management

Execution management aspects include locating, scheduling, aggregating, and managing the resources needed for multi-component jobs running in widely distributed environments.

- **global shells and generalized (e.g. global event driven) workflow management**
- **resource discovery and brokering**
  (locating all potential candidates for inclusion in a problem-specific virtual system based on a specification of problem resource requirements)
- **global queuing and execution queue management**

Grid Common Services: Execution Management (cont.)

- **distributed application management**
- **application-level fault management**
- **establishing the execution environment**

Grid Common Services: Runtime Support

- **Globus providing initial runtime services**
- **Runtime "system" services (TBD)**
- **Global file system**
- **Distributed data management**
  - **uniform, high-speed, wide area access to tertiary storage systems (e.g., MCAT/SRB [18])**
  - **data location management (e.g., GASS [2] and DPSS [6])**

Grid Common Services: Runtime Support (cont.)

- **high-speed application access to data files: staging, caching, remote I/O, and network caches as a service for applications**
- **global naming**
- **information oriented I/O mechanisms (data model and metadata based access)**
- **data cataloguing and publication services**
- **support for object oriented data management systems**

Grid Common Services: Runtime Support (cont.)

- **Comprehensive monitoring for bottleneck analysis, adaptation, and fault management**
- **Global event generation, publication, subscription, and management**
- **Checkpoint / restart**
- **Access control as a Grid service**
- **Services supporting collaboration and remote instrument management**
- **Application performance monitoring**

Grid Common Services: Environment Management

These are the core infrastructure services, supported by 7x24 operations staff, that make the Grid a reliable, production environment.

- **Grid Information Service (Globus MDS [2]) (the repository / publisher of “global” state information)**
  - **information needed to schedule jobs across the Grid: resource configuration, user allocations, ...**
  - **information needed to track jobs in the Grid**

Grid Common Services: Environment Management (cont.)

- **Grid security infrastructure (GSI and GAA [2]) (support for single sign-on without cleat text passwords, authenticated access to resources)**
  - **X.509 identity certificates from multiple CAs [22]**
  - **publication of identity certs**
  - **delegation of authority**
  - **general access control service**
- **Autonomous resource monitoring, management, and fault recovery**
- **Auditing (standard data model and XML [21] schema)**
  - **accounting, security, data change management**

Resources / Local Services

Most “resources” are “local” and will have their own resource managers and use policies. It is the usage mechanisms and interfaces for the local resources that the Grid common services are intended to homogenize. Such local services include:

- **CPUs and their batch queuing systems**
- **tertiary storage systems**
- **instrument control systems**
- **network QoS**

A key Grid issue for local resource management is support for co-scheduling and reservation local resources:

- CPU advance reservation
- network bandwidth advance reservation
- tertiary storage advance reservation and tape marshaling

Access Control and Security

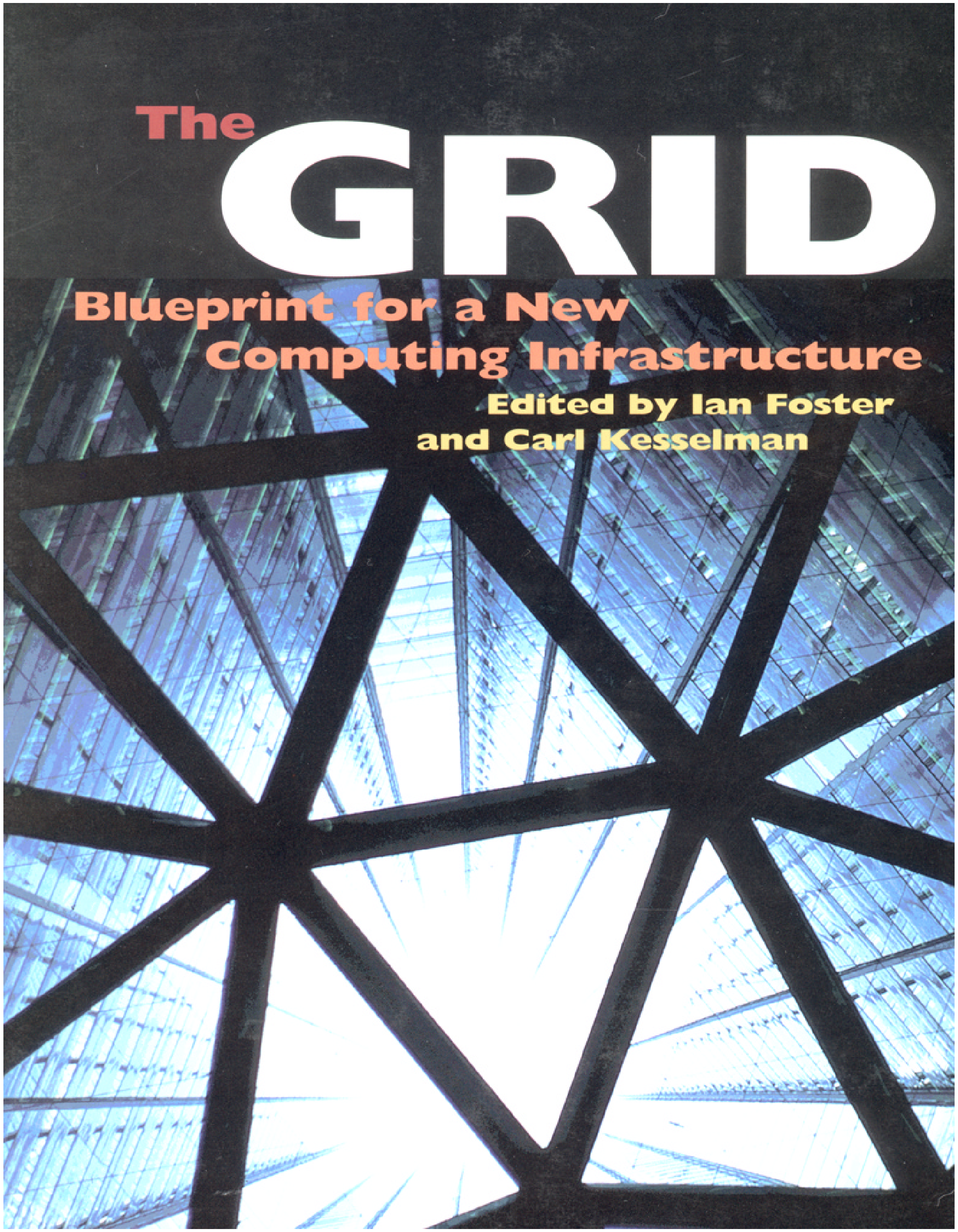
A security model for widely distributed systems that is supported by the security services noted above must address:

- **cyber risk mitigation and cross-site integrity**
- **control channel integrity and confidentiality**
- **optional data channel integrity and confidentiality**
- **identity management, authentication, and single identity sign-on w/o clear text passwords**
- **authorization via policy-based access control**
- **infrastructure assurance**

Internet / Telecommunications (DOE environment)

- **ESNet network infrastructure will support the DOE Science Grid distributed computing testbed and bandwidth reservation R&D.**

- **ESNet highspeed R&D links and DARPA NGI “Supernet” testbed (mixed optical WDM @ 10 Gb/s and SONET OC-48 - 2.5 Gb/s) will provide the DOE Science Grid high data-rate testbed.**



This book [3] examines some of the technologies and issues for computational and data Grids.

For ongoing work in this area see www.gridforum.org

References and Acronyms

[1] The Institute of Electrical and Electronic Engineers International Symposium on High Performance Distributed Computing (HPDC) provides a forum for presenting the latest research findings that unify parallel and distributed computing. In HPDC environments, parallel or distributed computing techniques are applied to the solution of computationally intensive applications across networks of computers.

[2] Globus is a middleware system that provides a suite of services designed to support high performance, distributed applications. Globus provides:

- Resource Management: Components that provide standardized interfaces to various local resource management systems (GRAM) manage allocation of collections of resources (DUROC). All Globus resource management tools are tied together by a uniform resource specification language (RSL).
- Remote Access: Components that enable remote access to files (GASS and RIO) and executables (GEM).
- Security: Support for single sign-on, authentication, and authorization within the Globus system (GSI) and (experimentally) authorization (GAA).
- Fault Detection: Basic support for building fault detection and recovery into Globus applications.
- Information Infrastructure: Global access to information about the state and configuration of system components of an application (MDS).

- Grid programming services: Support writing parallel-distributed programs (MPICH-G), monitoring (HBM), etc.

www.globus.org provides full information about the Globus system.

[3] *The Grid: Blueprint for a New Computing Infrastructure*, edited by Ian Foster and Carl Kesselman. Morgan Kaufmann, Pub. August 1998. ISBN 1-55860-475-8. http://www.mkp.com/books_catalog/1-55860-475-8.asp

[4] “Grids as Production Computing Environments: The Engineering Aspects of NASA's Information Power Grid,” William E. Johnston, Dennis Gannon, and Bill Nitzberg. Eighth IEEE International Symposium on High Performance Distributed Computing, Aug. 3-6, 1999, Redondo Beach, California. (Available at http://www.nas.nasa.gov/~wej/IPG)

[5] See www.nas.nasa.gov/IPG for project information and pointers.

[6] Tierney, B. Lee, J., Crowley, B., Holding, M., Hylton, J., Drake, F., “A Network-Aware Distributed Storage Cache for Data Intensive Environments”, Proceeding of IEEE High Performance Distributed Computing conference (HPDC-8), August 1999.

[7] “Real-Time Generation and Cataloguing of Large Data-Objects in Widely Distributed Environments,” W.Johnston, Jin G., C. Larsen, J. Lee, G. Hoo, M. Thompson, and B. Tierney (LBNL) and J. Terdiman (Kaiser Permanente Division of Research). Invited paper, International Journal of Digital Libraries - Special Issue on “Digital Libraries in Medicine”. May, 1998. http://www-itg.lbl.gov/WALDO/

[8] MAGIC: “The MAGIC Gigabit Network.” See: http://www.magic.net

[9] TerraVision-2: VRML based data fusion and browsing. (MAGIC consortium, NCAR, and NAVO: http://www.ai.sri.com/TerraVision/)

[10] Clipper: The goal of the Clipper project is software systems and testbed environments that result in a collection of independent but architecturally consistent service components that will enhance the ability of applications and systems to construct and use widely distributed, high-performance data and computing infrastructure. Such middleware should support high-speed access and integrated views for multiple data archives; resource discovery and automated brokering; comprehensive real-time monitoring and performance trend analysis of the networked subsystems, including the storage, computing, and middleware components, and; flexible and distributed management of access control and policy enforcement for multi-administrative domain resources. See http://www-itg.lbl.gov/~johnston/Clipper

[11] Legion is an object-based, meta-systems software project at the University of Virginia. http://www.cs.virginia.edu/~legion/

[12] The Grid Forum (www.gridforum.org) is an informal consortium of institutions and individuals working on wide area computing and computational grids: the technologies that underlay such activities as the NCSA Alliance's National Technology Grid, NPACI's Metasystems efforts, NASA's Information Power Grid, DOE ASCI's DISCOM program, and other activities worldwide.

[13] CAS

NASA HPCC Computational Aerosciences (CAS) Project

"The Computational Aerosciences (CAS) Project is one of five project areas within the NASA High-Performance Computing and Communications Program (HPCC). The CAS Project's primary goal is to accelerate the availability of high-performance computing hardware and software to the United States aerospace community for use in their design processes. The resulting capability will provide U.S. industry with a competitive advantage by reducing product cost and design cycle times. It is a complementary goal to hasten the emergence of a viable commercial market for hardware and software vendors to exploit this lead."

"The solutions of representative design problems, called CAS "Grand Challenge" applications, are used for guiding the development of system hardware and software. Although some of the algorithms and methods needed to eventually solve the CAS applications will find use in the aerospace industry, the primary purpose and product of the work leading toward CAS application solutions is to focus and guide computer system hardware and software development."

"This CAS project targets advances in aeroscience algorithms and applications, system software, and computing machinery that will enable more than 10,000-fold increases in system performance early in the Twenty-first Century. These computational capabilities will be sufficiently characterized such that they can be rapidly integrated into economical design and development processes for utilization by U.S. industry."

http://cas.arc.nasa.gov/, Cathy Schulbach project manager

[14] The Data Assimilation Office (DAO) at the NASA Goddard Space Flight Center, Greenbelt, Maryland, is dedicated to advancing the state of the art of data assimilation and to using the assimilated data in a wide variety of Earth system problems. See http://dao.gsfc.nasa.gov/. Also see http://www.nas.nasa.gov/~cheung/PROJECTS/DAO_Projects/DAO_Projects.html

[15] "The ISE Goal: To develop the capability for scientists and engineers to work together in a virtual environment, using simulation to model the complete life-cycle of a product/mission before commitments are made to produce physical products." www.ise.nasa.gov

[16] Astrobiology is a mutli-institutional NASA program seeking to identify the origins of life throughout the universe. http://astrobiology.arc.nasa.gov/

[17] The Portable Batch System (PBS) is a batch queueing system developed at NAS. PBS implements the POSIX standard, and operates on networked, multi-platform UNIX environments, including heterogeneous clusters of workstations, supercomputers, and massively parallel systems. PBS is the basis of the IPG global queuing system work. http://parallel.nas.nasa.gov/Parallel/PBS/

[18] Storage Resource Broker (SRB) provides uniform access mechanism to diverse and distributed data sources. The SRB provides the protocol conversion required to interface to heterogeneous data sources. The SRB interfaces with the MCAT metadata catalog to access metadata information about individual files and objects. http://www.sdsc.edu/MDAS/

[19] Alliance / NCSA: National Computational Science Alliance. “The National Computational Science Alliance (Alliance) is [an NSF funded] partnership among computational scientists, computer scientists, and professionals in education, outreach, and training at more than 50 U.S. universities and research institutions working to prototype the computational and information infrastructure of the next century.” www.ncsa.edu

[20] NPACI: The National Partnership for Advanced Computational Infrastructure. “Led by UC San Diego, NPACI [an NSF funded partnership] will revolutionize the nation’s computational infrastructure by building on the foundation of SDSC and involving 37 of the nation's leading academic and research institutions, located in 18 states from coast to coast. Driven by real applications needs, NPACI is developing software infrastructure to link the highest performance computers, data servers, and archival storage systems to enable easier use of the aggregate computing power.” www.npaci.edu

[21] “XML is the `Extensible Markup Language' (extensible because it is not a fixed format like HTML). It is designed to enable the use of SGML on the World Wide Web. XML is not a single, predefined markup language: it’s a metalanguage -- a language for describing other languages -- which lets you design your own customized markup languages for different classes of document.)” http://www.ucc.ie/xml

[22] PKI: Public-Key certificate Infrastructure. Public-key cryptography involves two keys, whereby data encrypted with one key can only be decrypted with the other, and visa versa. In PKI one key (the public-key) is freely available and the other is kept private. In this way, material encrypted with the private key and decrypted with the public-key proves that the holder of the private key must have been the originator of the material. A certification authority generates a certificate containing the name (usually X.500 distinguished name) of an entity (e.g. user) and that entity's public key. The CA then signs this "certificate" and publishes it (usually in an LDAP directory service). These are the basic components of PKI, and allow the entity to prove its identity, independent of location or system, by signing a token with the private key, handing the signed token to a system (e.g. as part of a login process), and then that system can verify the signer's identity by obtaining the identity certificate, extracting the entity's public key, and verifying the signature. The identity certificate (most commonly an X.509 certificate) is, in turn, verified by obtaining the CA's public key and using it to verify the contents.This later process is called digital signature and is accomplished by the certificate originator generating a unique hash of the certificate contents, and then encrypting that hash with the originator's private key. The hash is then appended to the certificate (or any other document) and may be used to both verify the originator's identity and the integrity of the contents (the hash function produces a "unique" hash for every byte string). For more information, see, e.g., RSA Lab's "Frequently Asked Questions About Today's Cryptography" http://www.rsa.com/rsalabs/faq/,

Computer Communications Security: Principles, Standards, Protocols, and Techniques. W. Ford, Prentice-Hall, Englewood Cliffs, New Jersey, 07632, 1995, or *Applied Cryptography*, B. Schneier, John Wiley & Sons, 1996.

[23] CoSMO: Consolidated Supercomputing Management Office - NASA's production supercomputing organization. Most CoSMO facilities are located and, and operated by, the NAS Division at Ames Research Center. http://www.cosmo.nasa.gov/

[24] Condor is a High Throughput Computing environment that can manage very large collections of distributively owned workstations. The environment is based on a novel layered architecture that enables it to provide a powerful and flexible suite of Resource Management services to sequential and parallel applications. http://www.cs.wisc.edu/condor/

[25] NREN: NASA Research and Education Network. NREN is an integral partner in IPG, and supplies the IPG network testbeds and does network bandwidth reservation R&D. http://www.nren.nasa.gov

[26] RLV: Reusable Launch Vehicle. NASA's next generation space shuttle design program.

[27] SCIRun is a scientific programming environment that allows the interactive construction, debugging and steering of large-scale scientific computations. SCIRun can be used for interactively:

- Changing 2D and 3D geometry models (meshes).
- Controlling and changing numerical simulation methods and parameters.

- Performing scalar and vector field visualization.

SCIRun uses a visual programming dataflow system. SCIRun is extensible to a variety of applications and will work with third party modules written in Fortran, C, and C++. http://www.cs.utah.edu/~sci/software/

[28] "The Network Weather Service is a distributed system that periodically monitors and dynamically forecasts the performance various network and computational resources can deliver over a given time interval." See http://nws.npaci.edu/

[29] "The NetLogger Methodology for High Performance Distributed Systems Performance Analysis," Brian Tierney, W. Johnston, J. Lee, G. Hoo, C. Brooks, D. Gunter. 7th IEEE Symposium on High Performance Distributed Computing, Chicago, Ill. July 29-31, 1998.

Netlogger provides services and tools for precision monitoring of application performance and communications, as well as related system and network events. http://www-didc.lbl.gov/NetLogger/

[30]