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

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**

Requirements (cont.)

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.

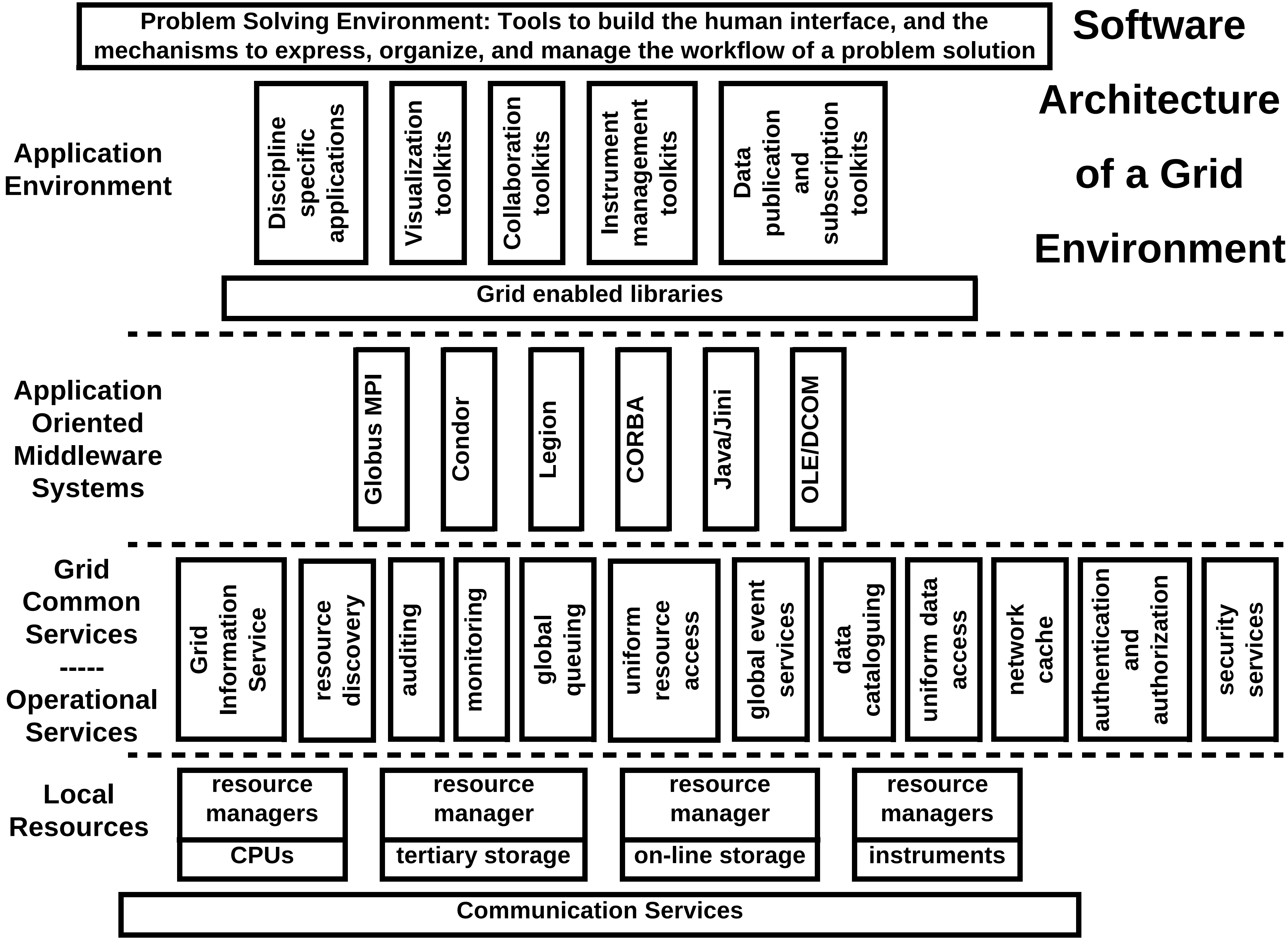
There is an additional set of requirements from the tool developers – the application domain computational scientists – primarily in the area of supporting code development and execution environment access and management. These will not be discussed here, but are also major drivers for developing Grid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

Software Architecture of a Grid Environment



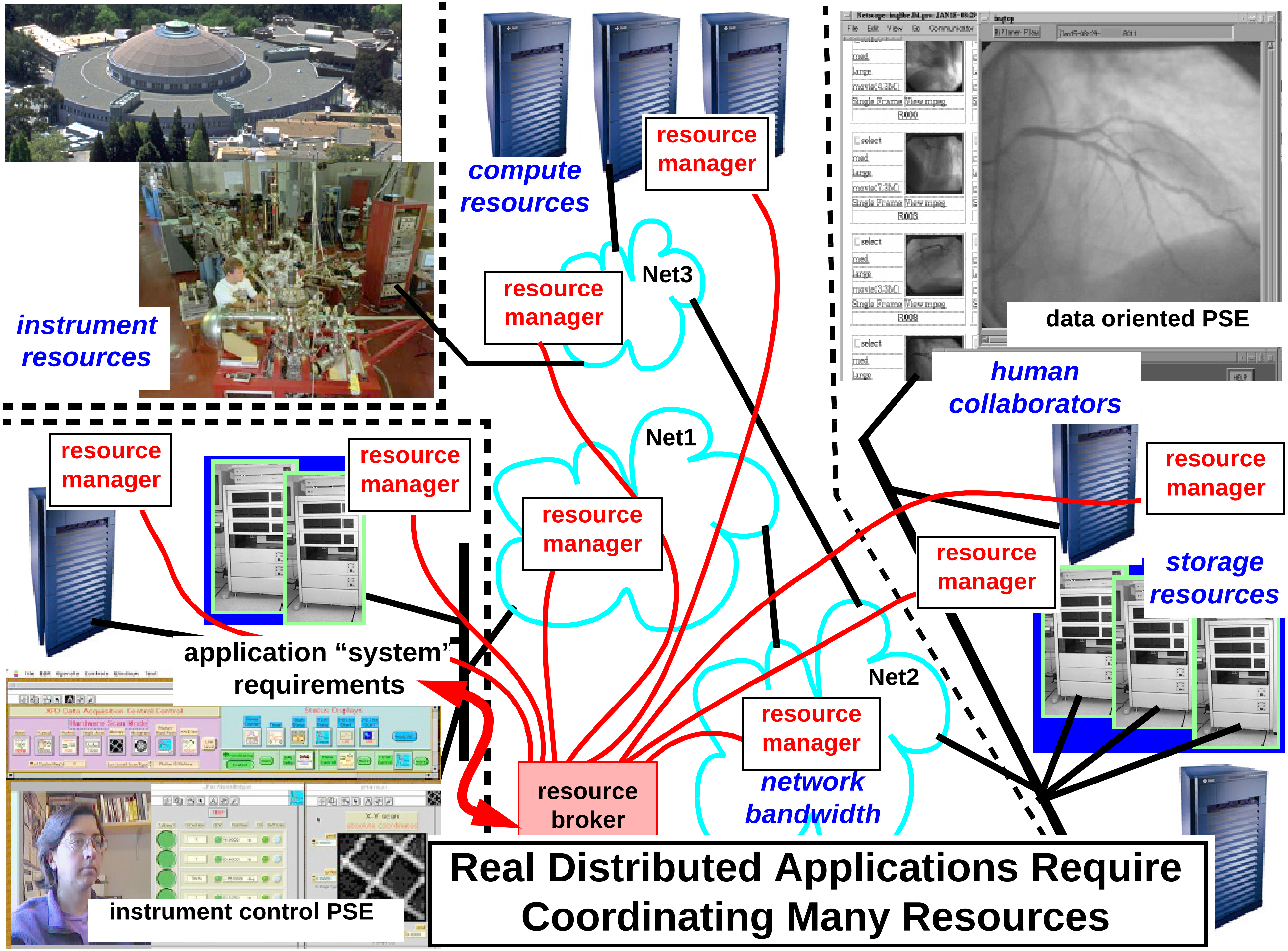
Vision for Science Grids

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.

Scale in Grids (cont.)

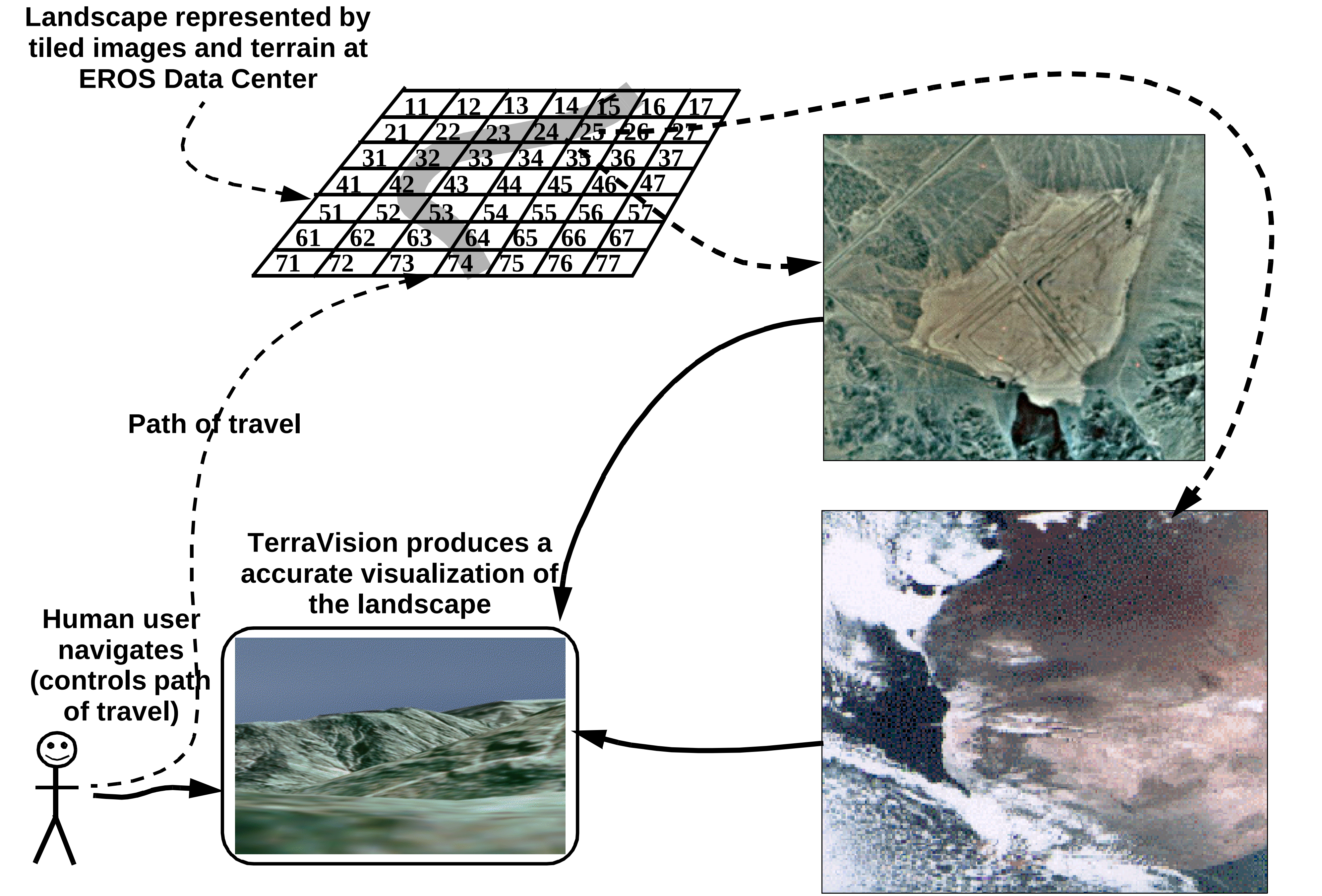
- **Coupling large-scale computing and data systems to scientific and engineering instruments requires that many heterogeneous resources be located, co-scheduled, and managed, including collaborators, instruments, bandwidth, and storage and computational systems**
  - **E.g.: real-time interaction with 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

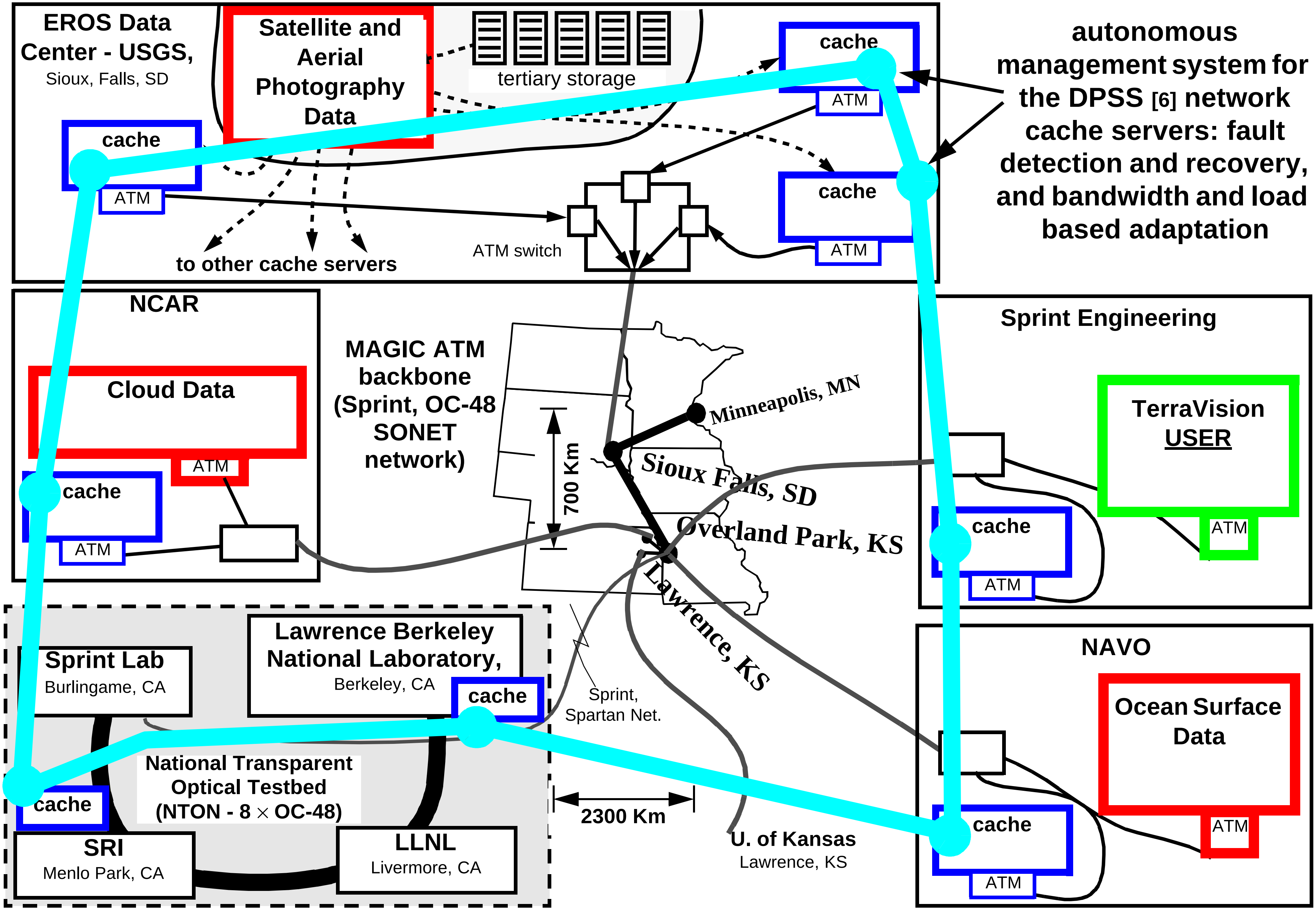


Scale in Grids - Data Intensive Computing (cont.)

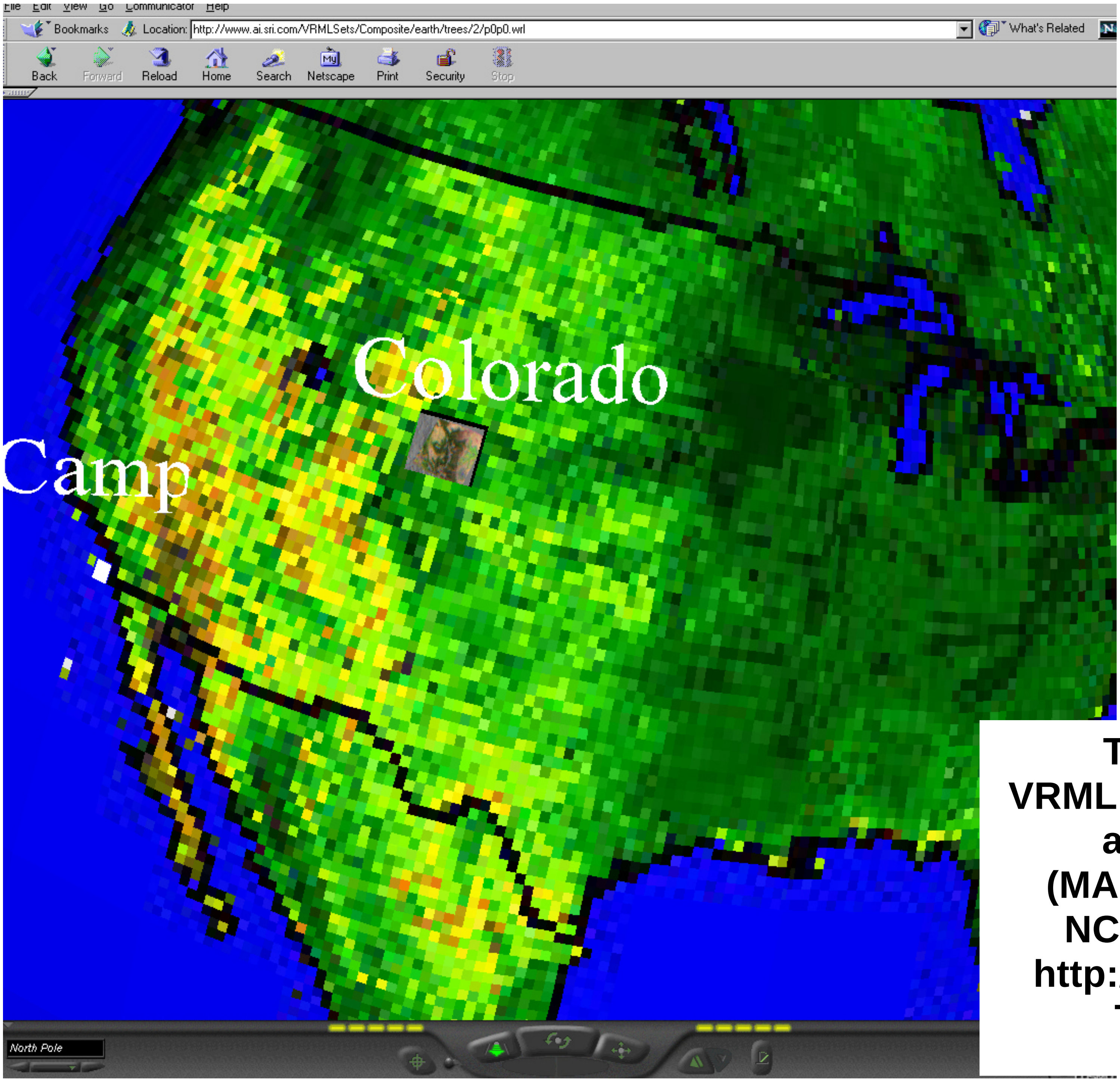
- **Example: High data-rate distributed data management and federated access for archived satellite and aerial imagery, digital terrain data, and atmospheric data ([8] and [9])**
  - **on-line, real-time access to multiple environmental data sets that are (and always will be) maintained by domain experts 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

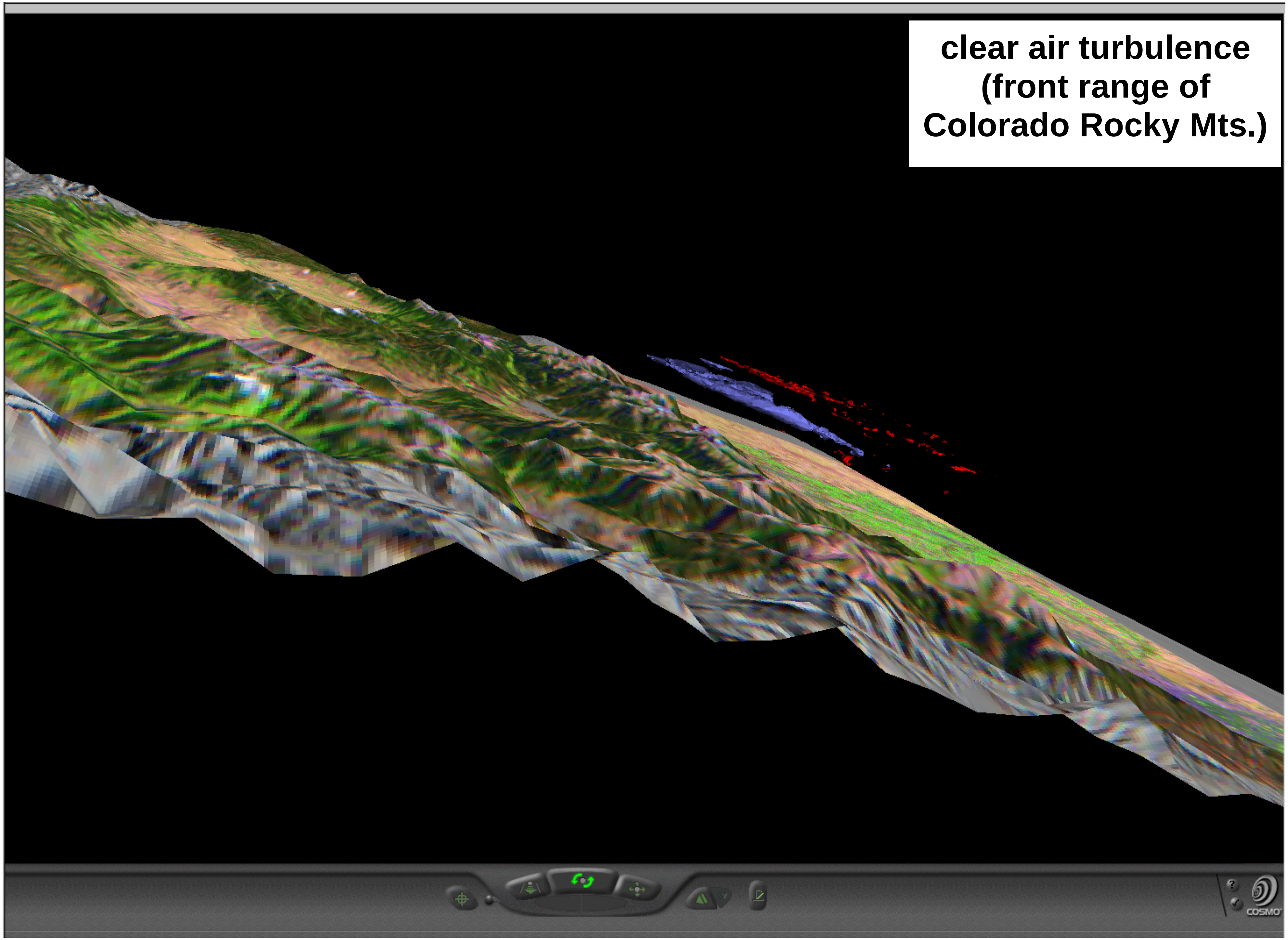


The MAGIC Testbed Distributed Application Environment



**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.)



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**
  - $\Rightarrow$ **the "user interface to the Grid"**
  - $\Rightarrow$ **knowledge based workflow control (required for: fault management ("FM") and extremely remote systems ("ERS"))**
- **High throughput job managers**
  - $\Rightarrow$ **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, fault detection and recovery, 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.)

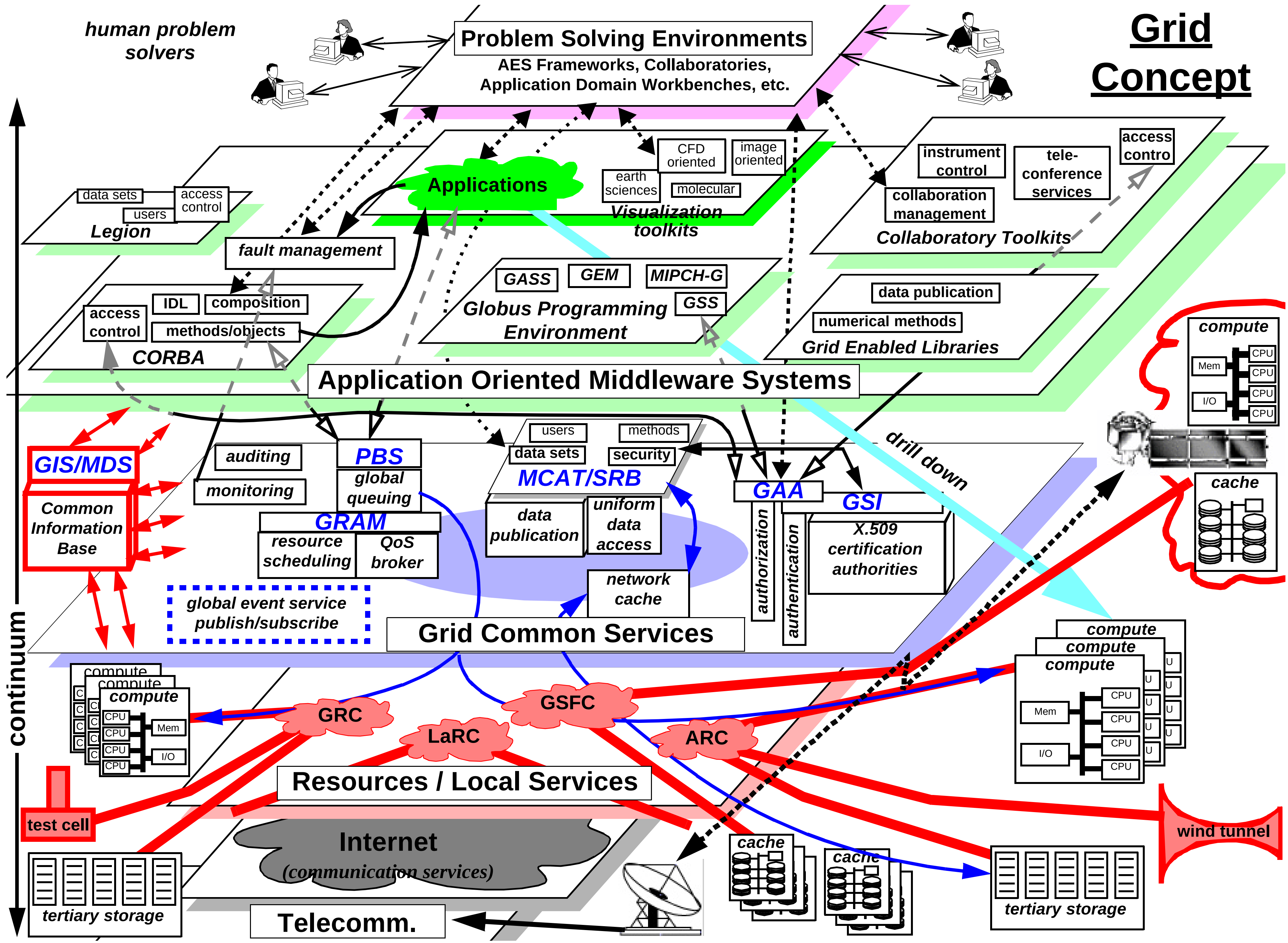
- **Global event management**
  - ⇒ **synchronization of widely distributed processes and data sets to support consistency/repeatability of results, and use of “independent” data sources**
  - ⇒ **primary information source for workflow management (FM & ERS)**
- **Grid enabled / aware algorithms**
 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

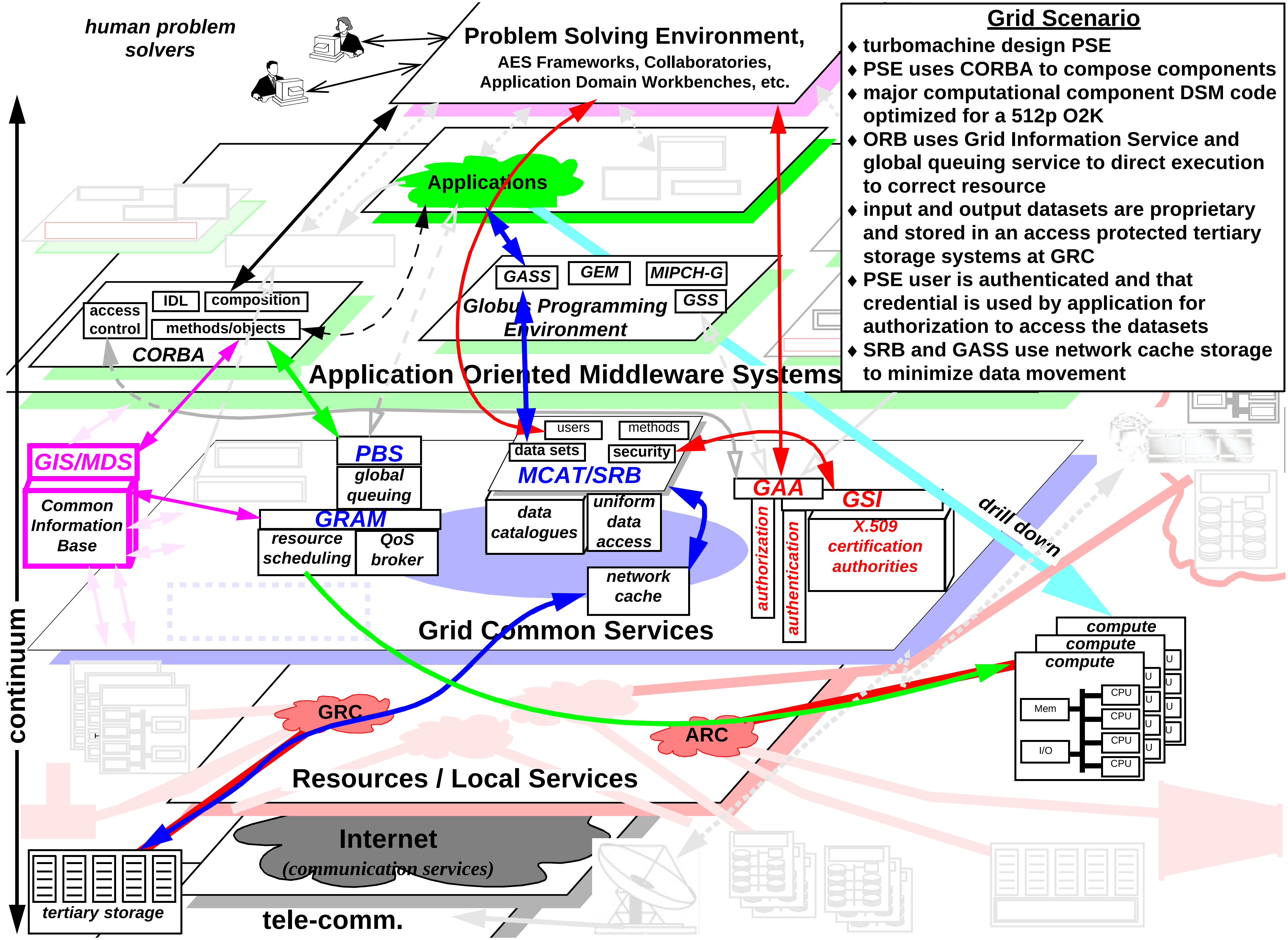
What Must Grids Provide (cont.)

- **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
human problem solvers
Problem Solving Environments
AES Frameworks, Collaboratories, Application Domain Workbenches, etc.
Applications
CFD oriented
image oriented
earth sciences
molecular
Visualization toolkits
instrument control
tele-conference services
access contro
collaboration management
Collaboratory Toolkits
data sets
access control
users
Legion
fault management
access control
IDL
composition
methods/objects
CORBA
GASS
GEM
MIPCH-G
GSS
Globus Programming Environment
data publication
numerical methods
Grid Enabled Libraries
Application Oriented Middleware Systems
compute
CPU
Mem
I/O
drill down
cache
GIS/MDS
Common Information Base
auditing
monitoring
PBS
global queuing
GRAM
resource scheduling
QoS broker
users
methods
data sets
security
MCAT/SRB
data publication
uniform data access
network cache
GAA
GSI
X.509 certification authorities
authorization
authentication
global event service publish/subscribe
Grid Common Services
GRC
GSFC
LaRC
ARC
Resources / Local Services
test cell
wind tunnel
Internet
(communication services)
tertiary storage
Telecomm.
continuum



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