Information Power Grid

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

William E. Johnston,

Dennis Gannon, and Bill Nitzberg

Numerical Aerospace Simulation Division

William J. Feiereisen, Division Chief,

William Thigpen, Engineering Branch Chief,

Alex C. Woo Research Branch Chief

http://www.nas.nasa.gov/IPG

Prototype and Testbed Environments

The IPG *prototype production* environment (the "CX" IPG testbed) targets an *operational Grid environment incorporating major computing and data resources at multiple NASA sites* in order to provide an infrastructure capable of routinely addressing larger scale, more diverse, and more transient problems than is possible today.

What is the Information Power Grid?: Prototype (cont.)

The CX testbed will address aggregated large system computing and data management:

- **CPU resource reservation**
- **network resource reservation**
- **standardized access to high capacity storage**
- **a stakeholder-oriented, distributed resource management approach that addresses *global use but local control***
- **operational tools and procedures and user support**
- **allocation, accounting, and auditing**
- **strong user identification and access control**
- **model for CoSMO [23]**

What is the Information Power Grid?: Prototype (cont.)

IPG testbeds will also provide R&D platforms:

- ***Dev*** **testbed: IPG system software development**
- ***DX*** **testbed: high data-rate distributed resources**
  - **high-speed end-to-end**
  - **high data-rate services including data archives and instrument systems**
  - **test applications**
- ***SX*** **testbed: security and infrastructure protection**

What is the Information Power Grid? (cont.)

Research and development in support of long term goals

(see ‘ *“Top Twenty” R&D topics for IPG*,’ below)

How is IPG Being Accomplished?

¨ **NASA: IT/ACNS, HPCC/CAS, and HPCC/NREN**

¨ **Collaboration between**

- **several NASA centers (Ames/NAS, GRC, LaRC, GSFC)**
- **the NSF PACIs (NCSA / Alliance [19] and SDSC / NPACI [20])**
- **universities and government labs**

¨ **Areas that must be addressed for baseline IPG:**

1) **persistent operational environment that encompasses significant resources**
2) **new service delivery / operational models**
3) **new functionality**

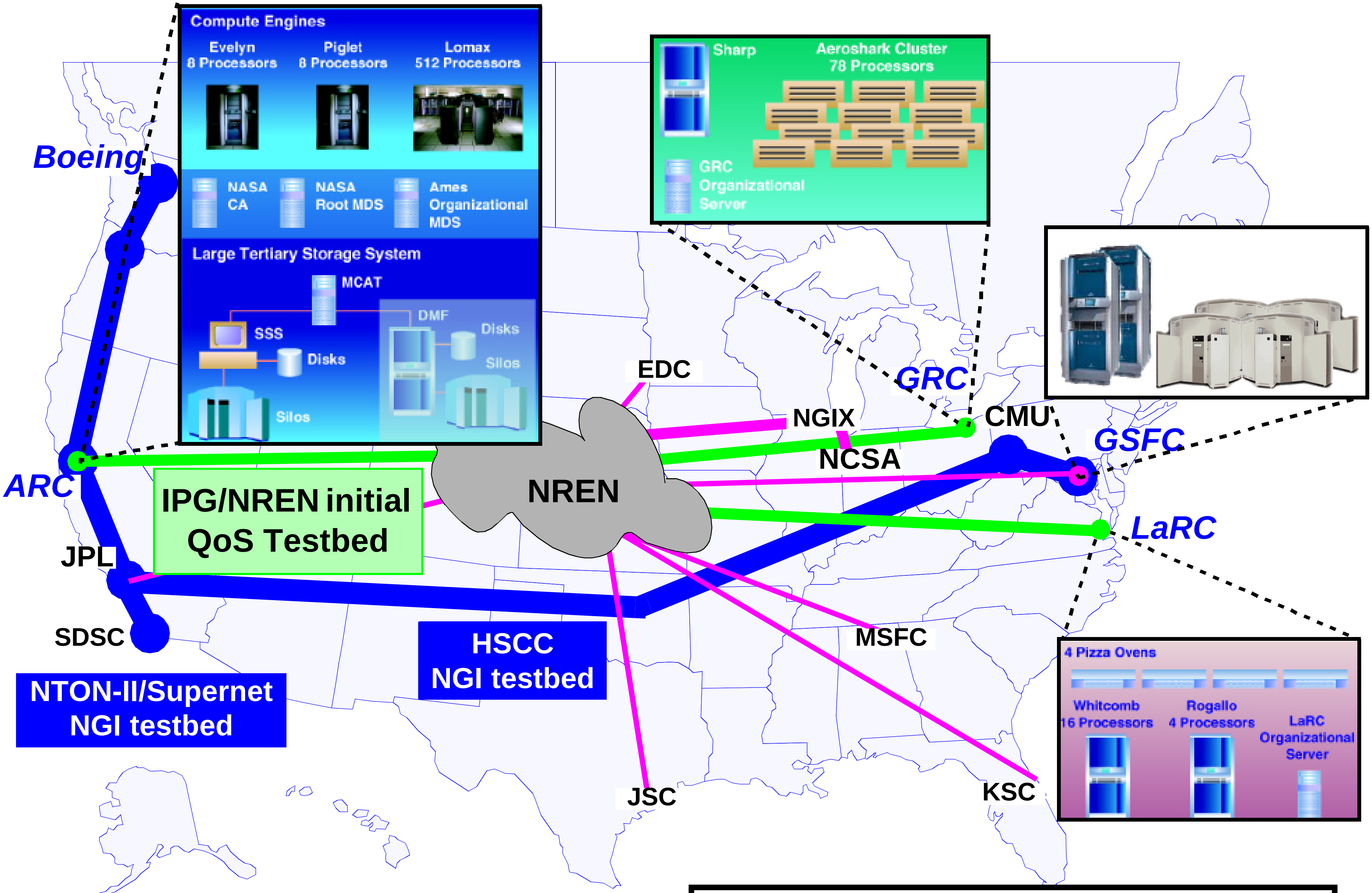
What is the State of IPG?

11/99 Baseline Operational System

- **Computing resources: ≈600 CPU nodes in half a dozen SGI Origin 2000s and several workstation clusters at Ames, Glenn, and Langley, with plans for incorporating Goddard and JPL**

- **Wide area network interconnects of at least 100 mbit/s**

- **Storage resources: 30-100 Terabytes of archival information/data storage *uniformly* accessible from all IPG systems via SDSC’s MCAT and SRB**



Compute Engines

Evelyn
8 Processors

Piglet
8 Processors

Lomax
512 Processors

NASA
CA

NASA
Root MDS

Ames
Organizational
MDS

Large Tertiary Storage System

MCAT

SSS

Disks

Silos

DMF

Disks

Silos

Sharp

Aeroshark Cluster
78 Processors

GRC
Organizational
Server

4 Pizza Ovens

Whitcomb
16 Processors

Rogallo
4 Processors

LaRC
Organizational
Server

Boeing

ARC

JPL

SDSC

EDC

NGIX

NCSA

NREN

GRC

CMU

GSFC

LaRC

MSFC

JSC

KSC

IPG/NREN initial
QoS Testbed

HSCC
NGI testbed

NTON-II/Supernet
NGI testbed

IPG Baseline System and High Data-Rate (DX) Testbed

State of IPG: Baseline Operational System (cont.)

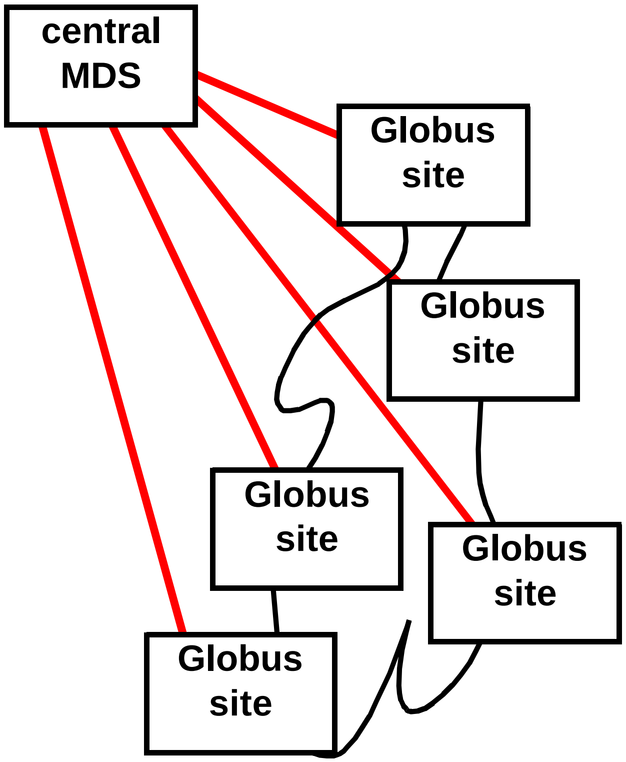
- **Globus providing the Grid common services**
- **Programming and program execution support**
  - **Grid MPI (via the Nexus communications library)**
  - **CORBA integrated with Globus**
  - **global job queue management**
  - **high throughput job manager**
  - **Condor [24] ("cycle stealing" computing)**
- **A stable and supported operational environment**
- **Several "benchmark" applications operating across IPG (multi-grid CFD code, parameter study)**
- **Multi-Grid operation (applications operating across IPG and NCSA)**

How Are We Accomplishing the Baseline?

1) Persistent operational environment that encompasses significant resources

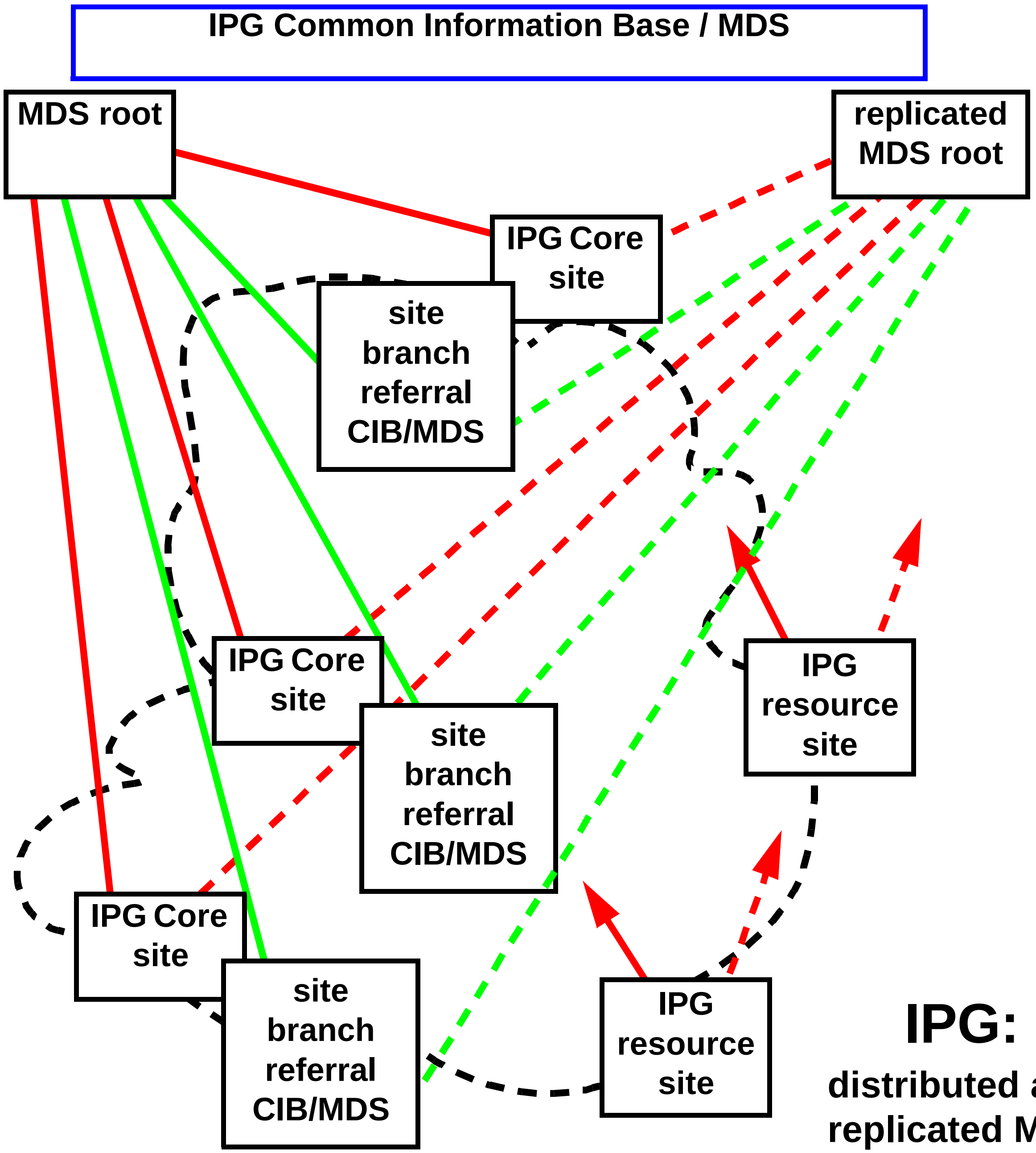
¨ **“IPG Prototype Startup Tasks (target: 6/99)” (Section B.3)**

- **“Globus deployed across Ames, GRC, and LaRC (Task 1.0)” (Section B.3.1)**
- **“IPG “common grid information base” (Task 2.0)” (Section B.3.2)**
- **“IPG X.509 Certification Authority and certificate server (Task 3.0)” (Section B.3.3)**
- **““Global” queuing and user-level queue management capability on top of Globus (Task 4.0)” (Section B.3.4)**



Gusto Testbed:

central MDS


IPG Common Information Base / MDS
MDS root
replicated MDS root
IPG Core site
site branch referral CIB/MDS
IPG Core site
site branch referral CIB/MDS
IPG Core site
site branch referral CIB/MDS
IPG resource site
IPG resource site

IPG:

distributed and replicated MDS

Migration of Common Information Base from R&D to Prototype Production

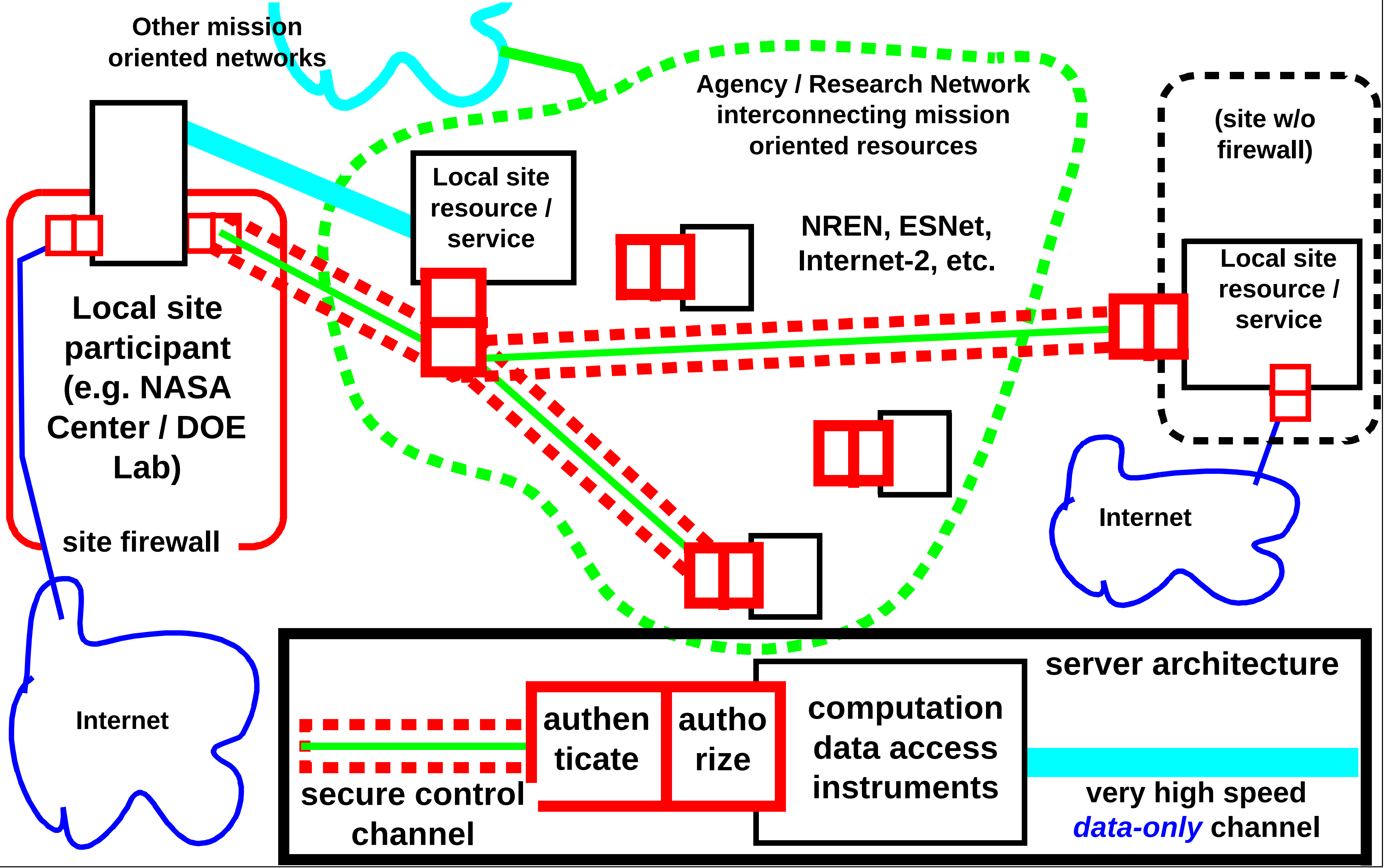
Accomplishing the Baseline - Operational Environment

- **“Computing resources for the initial IPG multi-center testbed (“CX”) (Task 5.0)” (Section B.3.5)**
- **“Networking for the IPG Testbed: Ames, GRC, LaRC (Task 6.0)” (Section B.3.6)**
- **“IPG Access for Archival and Published Data: SDSC’s Metadata Catalogue (MCAT) and the Storage Resource Broker (SRB) (Task 7.0)” (Section B.3.7)**
- **“Heterogeneity in the IPG testbed: Condor (Task 8.0)” (Section B.3.8)**
- **“Heterogeneity in the IPG testbed: High performance clusters (Task 9.0)” (Section B.3.9)**

Accomplishing the Baseline - Operational Environment

- **“IPG Operational Tasks (needed for v.1.0 of IPG baseline - 9/99)” (Section B.4)**
  - **“Security (Task 10.0)” (Section B.4.1)**
  - **“IPG Information Base / MDS database maintenance (Task 11.0)” (Section B.4.2)**
  - **“IPG/Globus system administration (Task 12.0)” (Section B.4.3)**
  - **“Automatic monitoring of IPG components (Task 13.0)” (Section B.4.4)**
  - **“Trouble ticket model (Task 14.0)” (Section B.4.5)**
  - **“Condor support (Task 15.0)” (Section B.4.6)**
  - **“CORBA support (Task 16.0)” (Section B.4.7)**
  - **“Legion support (Task 17.0)” (Section B.4.8)**
  - **“Documentation (Taks 18.0)” (Section B.4.9)**

Security Model: All command and control functions are transported over an encrypted channel, after the client/user is authenticated and authorized. This compartmentalizes all servers: If multiple servers are involved in a distributed system, then each reauthorizes connections through the use of cryptographic proxies or active re-authentication.



Accomplishing the Baseline - Operational Environment

- **“User services (Task 19.0)” (Section B.4.10)**
- **“Account management (automated generation and maintenance mechanisms) (Task 20.0)” (Section B.4.11)**
- **“Globus with multiple MDS and PKI (Task 21.0)” (Section B.4.12)**
- **“Allocation Management and Accounting (Task 22.0)” (Section B.4.13)**
- **“System testing: Verification suites, benchmarks, and reliability/sensitivity analysis for IPG (both static and dynamic) (Task 23.0)” (Section B.4.14)**

Accomplishing the Baseline - Operational Environment

·· "IPG Functionality Tasks (near-term - 9/99)" (Section B.5)

- "CORBA in the IPG environment. (Task 24.0)" (Section B.5.1)
- "Integration of Legion (Task 25.0)" (Section B.5.2)
- "CPU resource reservation (Task 26.0)" (Section B.5.3)
- "High Throughput Computing (Task 27.0)" (Section B.5.4)
- "Programming Services (Task 28.0)" (Section B.5.5)
- "Distributed debugging (Task 28.1)" (Section B.5.6)
- "Grid enabled visualization (Task 28.2)" (Section B.5.7)

Accomplishing the Baseline - Operational Environment

- “IPG Functionality Tasks (mid-term - 9/00)” (Section B.6)
    - “Network bandwidth reservation (Task 29.0)” (Section B.6.1)
- “Characteristic Applications - 9/99” (Section B.7)
    - “OVERFLOW port & tune (Task 31.0)” (Section B.7.1)
    - “NPSS port & tune (Task 32.0)” (Section B.7.2)
    - “Parameter study” (Section B.7.3)
    - “Heterogeneous testbed application: Condor (Task 34.0)” (Section B.7.4)
    - “Heterogeneous testbed application: High performance clusters (Task 35.0)” (Section B.7.5)
- “First Year IPG review (Task 36.0)” (Section B.8)

How Are We Accomplishing the Baseline? (cont.)

2) New service delivery / operational models

Grids such as IPG, effectively define a new business model for operational organizations that will deliver large-scale computing and data resources *in ways that allow them to be integrated with other widely distributed resources controlled*, e.g., by the user community.

How Are We Accomplishing the Baseline? - Service Model (cont.)

Implementing this service model requires two things:

- ***tools for production support*** **of integrated collections of widely distributed resources must be identified, built, and provided to the systems and operations staffs**
- ***new organizational structures*** **must be evolved that account for the fact that operating Grids is different than operating traditional supercomputer centers, and management and operation of this new shared responsibility service delivery environment must be explicitly addressed**

How Are We Accomplishing the Baseline? (cont.)

3) New functionality will arise from:

- A *detailed and on-going examination of requirements* generated by several NASA application communities, both in terms of specific capabilities identified by the applications community
- Continual analysis of the requirements and desired operating environments by computer scientists
- Research and development to address the identified needs

The Two Year IPG Goals

- **An *operational and persistent, "large-scale" prototype-production Information Power Grid providing access to computing, data, and instrument resources at NASA Centers around the country***

- **A well developed R&D program addressing open issues in Grid computing and data management: i.e., defined tasks and progress in the areas listed under " *Research and development in support of long term goals*," above.**

- **Applications that cannot be done today:**
    - ***computational grid-independent rotocraft reference simulation***
    - ***multi-component, multi-disciplinary turbo-machine simulation***
    - ***configurable problem solving environment for large parameter space studies (e.g., RLV [26])***
    - ***on-line instrument (Darwin wind tunnel system) coupled to scheduled computational and data resources***

- **Transfer of Grid technology to the production environment (CoSMO)**

What Will IPG Facilitate?

¨ **Building just-in-time, large-scale systems/applications to support scientific and engineering *computing and data oriented activities* that are not steady state, i.e. those that may require, or have to make use of, a different resource mix for every different problem – e.g.:**

- ***coupled, multidisciplinary simulations***
- ***large-scale simulations***
- ***coordinated use of many dispersed data archives***
- ***data analysis based control of on-line instruments***
- ***coupling of remote, on-line instruments to large-scale computation simulation***

What Will IPG Facilitate? (cont.)

¨ ***Routine use of wide area, data-intensive applications***

- **remote access to high data-rate real-time data sources / instruments and very large datasets**

¨ ***Building and managing just-in-time, dynamic, and collaborative systems***

- **coordinated work by dispersed groups based, e.g., on central design databases (e.g. airframe or turobmachinery geometry and performance)**
- **data and simulation based crisis response**

“Top Twenty” R&D topics for IPG

These topics address functionality that is of primary important for some aspect of IPG in the first two years in order to satisfy an explicit requirement (either from user community or from operations/systems) or an implicit requirement (clearly needed to support our Grid model). This is not an exhaustive list of such topics because it excludes topics covered in the Engineering WG (functionality that is needed in the first year) and it does not cover topics that are being worked on by IPG funded partners (e.g. the NSF PACIs).

“Top Twenty” R&D topics for IPG (cont.)

¨ ***Tools for Converting and “Wrapping” Legacy Codes for Operation in Grids (section 7.2, page 111)*** (section C.3.4, page 433)

- **Most existing simulations are codes that have evolved over many years, together with the understanding of the underlying physics of the problem. As such, few of these codes are “modern” in design: single function code units, well documented and recoverable exceptions, re-entrant, etc. Yet these codes must be usable in modern distributed computing environments in order to make use of the available computing and storage resources.**

¨ ***Toolkits for Constructing Problem Solving Environments / Workbenches (section 6.1, page 103)*** (section C.2.1, page 432)

- **Provide the tools and techniques for integrating Grid applications and services into the desktop environment in ways that can express and facilitate the human work process. This includes: General graphical user interface components for providing the application interface; Scripting / PSE program preservation and editing; Interfaces to generalized workflow management services; Tools for managing high throughput jobs; Global shell” functionality; Rule-based workflows driven from published/subscribed events; Resource discovery and brokering services; Data cataloguing and access functions; Access control functions.**

- ***Application-Level Fault Management (section 8.6, page 131)*** (section C.4.6, page 435)
  - **Beyond the process-existence oriented monitoring and management approach of the previous section, in multi-element production software systems (such at DAO’s data acquisition, validation, and modeling for daily weather forecasting) application specific faults must also be dealt with. That is, mechanisms are required for managing generalized “faults” in all aspects of the application environment. The concept of generalized faults is intended to include things like application problems such as failure to achieve numerical convergence within a specified CPU time utilization, non-availability of resources or data, etc. These and other types of faults must be detected and acted upon to provide for recovery (continued operation of the application). The definition of these faults and the recovery procedures will be application specific, so what is needed is a framework that provides tools for fault definition and detection, followed by knowledge/rule based recovery.**

- ***Global Shells and Generalized Workflow Management (section 8.1, page 118)*** (section C.4.4, page 435)
  - **The Grid workflow management mechanism must provide for description and subsequent control of the ordered steps and events that represent a “job”. An approach analogous to shells in our local system environments might have Unix shell-like semantics, but global reach and naming. A more general approach is represented by a rule-based execution management system driven from published/subscribed global events (where the “events” represent process completion, file or other state creation, instrument turn-on, etc.).**

·· ***High-Speed Application Access to Data Files: Staging, Caching, Location Management, and Remote I/O (section 9.6.3, page 151)*** (section C.5.6.3, page 437)

- **Data location management is a very important facility in the Grid, and will probably be one of the services that most determines whether the Grid can truely be successful in providing widely distribtued access to computing, data, instrument, and human resources.**

·· ***Establishing the Execution Environment (section 8.2, page 123)*** (section C.4.3, page 435)

- **Tools are needed to set up the environment in which code must actually execute: binaries have to staged, runtime libraries have to be instantiated, file system environment must be established, environment variables and other state needed must be established. This situation presents both the problem of how to describe what is needed and then to establish all of the required environment on the target system.**

·· ***Application Performance Monitoring (section 9.8, page 160)*** (section C.5.8, page 438)

- **There are several aspects to monitoring. Characterizing the operating state and predicting future state of the execution environment is important for supporting execution management such as fault management and for adaptation and performance anaysis of running algorithms. Analytical monitoring is important for both algorithm characterization and troubleshooting. Characterization and**

prediction is provided, e.g., by the Network Weather Service [86] and characterization analysis is typified by the precision time event tagging for dispersed, multi-component performance analysis provided by Netlogger [85]. Further, monitoring is likely to be important for functionality such as generalized auditing e.g., data file history and control flow tracking in distributed, multi-process simulations. Work is needed on what should be done, and how existing approaches could be integrated into IPG.

- ***Global Queuing and Execution Queue Management (section 8.4, page 126)*** (section C.4.2, page 435)
  - **There are a wealth of queue management related issues and services that must be addressed in order to make Grids credible production environments and to provide better reliability and resource utilization than is available today. While some of these issues are arguably the purview of resource management, global queue management is resource independent, and is therefore included in execution management.**

- ***Support for Multiple Programming Paradigms in Multi-platform, Heterogeneous Computing Environments: Globus, CORBA, Condor, Java, and Legion (section 7.1, page 110)*** (section C.3.1, page 433)
  - **The most common programming paradigms identified thus far in the NASA environment are MPL/Fortran, MPI/Fortran, and Java. CORBA is used to wrap codes so that standardized interfaces may be defined. In addition to Globus [55], Java,**

Condor [37], and Legion [78] are actively used programming paradigms that will be supported in IPG. In the case of CORBA support means ensuring co-existence with IPG services and investigating and implementing cooperation with Grid services. In the case of Java support means interfaces to IPG services. Condor is already integrated. In the case of Legion, support in the IPG environment means that Legion will make use of the Grid Common Services such as the Common Information Base, job submission and management, etc.

¨ ***Network Bandwidth Advance Reservation Scheduling: Differentiated IP Services (section 11.2, page 171)*** (section C.7.2, page 440)

- **Reservable, high bandwidth data flows are essential to support transient network uses such as on-line scientific instruments that produce data only during experiments, specific data analysis exercises, or for large distributed computational runs when co-scheduling computing and communication resources will be necessary to keep the processors operating at a reasonable efficiency. These are scenarios in which an ensemble of data sources, and computing and storage resources must be connected and operated as a system — that is, all resources must be available at the same time as must the communications services the provide the interconnections to form useful distributed systems and to provide the “agility” to support transience (systems built on-demand for limited periods). Such reservation capability must be provided as a standard IPG service.**

¨ ***CPU Advance Reservation Scheduling (section 11.1, page 170)*** (section C.7.1, page 440)

- **The rationale for advance CPU reservation is the same as advance network bandwidth reservation: This is an essential capability to support building aggregated and just-in-time systems. This capability must be a standard IPG service.**

¨ ***High Throughput Computing / Parameter Study Support (section 6.3, page 104)*** (section C.2.1.1, page 432)

- **High-throughput computing / parameter study assigns independent tasks to multiple independent computing resources and schedules and manages them so that a large number of similar jobs execute in parallel. Sometimes a goal is to use otherwise unused computer cycles. Aerospace system design often requires searches through parameter space. In such cases, since each parameter is independent, individual or groups of parameters can be assigned to separate computers and then processed independently.**

¨ ***Data Cataloguing and Publication Services (section 9.6.4, page 152)*** (section C.5.6.4, page 437)

- **Services are needed to provide:**
  - **automatic characterization of datasets and “publication” of the characterization where, and in a form, that it may be located**
  - **description of data (generate metadata) and data formats**
  - **management of dataset use conditions and access control**

¨ ***Grid Distributed Debugging (section 7.4, page 114)*** (section C.3.3, page 433)

- **There currently are no commercial or experimental tools that can be directly incorporated into a Grid programming environment that can provide the needed level of support for distributed system debugging. Consequently, research and development is needed to adapt the current generation of parallel debuggers for dynamic and adaptive Grid applications.**

¨ ***Remote Visualization (section 6.6.2, page 106)*** (section C.2.3, page 432)

- **Provide the tools and techniques for integrating Grid applications and services into the desktop environment in ways that can express and facilitate the human work process. This includes: General graphical user interface components for providing the application interface; Scripting / PSE program preservation and editing; Interfaces to generalized workflow management services; Tools for managing high throughput jobs; Global shell" functionality; Rule-based workflows driven from published/subscribed events; Resource discovery and brokering services; Data cataloguing and access functions; Access control functions.**

¨ ***Access Control (section 13.7, page 203)*** (section C.8.2, page 441)

- **Access control is essential in order to address several issues in Grid environments. One obvious issue is that owners of confidential or proprietary data must have strong and easily managed mechanisms that ensure only authorized users gain access to that data. From the point of view of the Grid itself, management of**

stakeholder rights (use-conditions) is an essential aspect of having individuals and institutions make their resources available as part of any Grid environment. Many resources may only be utilized by people/communities that satisfy certain criteria as set by funding agencies, etc., as a matter of policy. Tools, techniques, and infrastructure must be built into IPG from the start in order to address these issues.

¨ ***Distributed Collaboration Tools and Techniques (section 6.4, page 104)*** (section C.2.2, page 432)

- **Toolkits supporting the construction of PSEs must provide the mechanisms for integrating computer mediated, distributed human collaboration into desktop problem solving environments. E.g. interface sharing, graphical user interface components that map to applications and Grid services, access control, a representation of the human work process that maps onto the workflow management mechanism, etc.**

¨ ***High-Speed Application Access to Data Files: Staging, Caching, Location Management, and Remote I/O (section 9.6.3, page 151)*** (section C.5.6.3, page 437)

- **Data location management is a very important facility in the Grid, and will probably be one of the services that most determines whether the Grid can truely be successful in providing widely distribtued access to computing, data, instrument, and human resources.**

Partners

NASA partners:

- Glenn Research Center
- Langley Research Center

Funded collaborations:

- NSF PACIs via NASA / NSF MOU (Alliance/NCSA and NPACI/SDSC)
- U. Ill., U. Chicago, U. Wisc., Argonne National Lab., UCSD, USC/ISI, U. Tenn., U. Va.

Others:

- Lawrence Berkeley National Lab (network QoS)