Technical Report NEESgrid-2003-13
www.neesgrid.org

(Whitepaper Version: 1.1)
Last modified: February 18, 2004

NEESgrid Requirements Traceability Matrix

Thomas A. Finholt, Dan Horn, Suzandeise Thomé [1]

[1] Collaboratory for Research on Electronic Work, School of Information, University of Michigan

Feedback on this document should be directed to danhorn@umich.edu

Acknowledgment: This work was supported primarily by the George E. Brown, Jr. Network for Earthquake Engineering Simulation (NEES) Program of the National Science Foundation under Award Number CMS-0117853.

Summary

This document presents the Requirements Traceability Matrix created by the User Requirement team and the methodology used to generate it. The Requirements Traceability Matrix is a representation of user requirements aligned against system functionality. It is used to ensure that all requirements are being met by the system deliverables.

The Requirements Traceability Matrix developed for the NEESgrid project indicates that 76% of the user requirements are being addressed by the system integration team and will be implemented in the first version of the system. 24% of the user requirements are not being addressed by the system integration effort. These numbers do not represent the effort spent on user requirements relative to each other. Each user requirement was considered as one unit, independently of the amount of system integration effort required for each one.

This document supersedes Requirements Traceability Matrix version 1.0.

Introduction

In the report from the March 2003 Site Visit Team, it was strongly recommended that the NEESgrid project complete a Requirements Traceability Matrix (RTM). This document presents the RTM created by the User Requirement team and the methodology used to generate it.

The Requirements Traceability Matrix is a representation of user requirements aligned against system functionality. It is used to ensure that all requirements are being met by the system deliverables. This technique is commonly used in large-scale government software development projects, such as the Center-TRACON Automation System, the National Polar-Orbiting Operational Environmental Satellite System (NPOESS), and the National Integrated Land System (NILS). It is now increasingly being used in commercial software development projects as well.

Methodology

The creation of the NEESgrid Requirements Traceability Matrix involved the following steps, which are explained in more detail in the pages that follow.

1. Identification of user requirements
2. Identification of system components
3. Estimation of effort spent on each system component
4. Mapping of system components to user requirements

1 - Identification of user requirements

The first step in the creation of a Requirements Traceability Matrix was the identification of user requirements. A list of user requirements was generated from the following NEESgrid documents:

- NSF Program Solicitation (http://www.nsf.gov/pubsys/ods/getpub.cfm?nsf007)
- NSF Site Visit Reports (from 2001, 2002 and 2003)
- System Overview (http://neesgrid.org/documents/NEESSystemOverview2_1.pdf)
- NEESgrid User Requirements Document v2.0 (http://www.neesgrid.org/documents/NEESgrid_UR_Feb15.2002.pdf)
- NEESgrid User Requirements Document v3.0 (http://neesgrid.org/documents/NEESgrid_User_Requirements_v3_0.pdf)

User requirements were then grouped into the following categories:

- Collect and Store data
- Search data
- Manage data
- Database Administration
- Hybrid experiments
- Data Viewing
- Synchronous collaboration
- Asynchronous collaboration
- Other Collaboration Tools - Synchronous and Asynchronous
- Simulation codes
- Repository Curation
- Access to high performance computing resources
- Security
- Safety
- Usability
- Network
- Performance
- System capacity
- Other system characteristics
- Support for future user constituents

2 - Identification of system components

The second step in the creation of the RTM was the identification of system components. The documents that served as input to the identification of system components are listed below.

- NEESgrid System Baseline Description v 0.9.5 (http://neesgrid.org/documents/SI_SBD_v09_5_03_03_2003.pdf)
- NEESgrid System Architecture v1.0 and v1.1 (http://neesgrid.org/documents/NEESgrid_SA_Feb15.2002.pdf, http://neesgrid.org/documents/NEESgrid_SystemArch_v1.1.pdf)

The following system components, or functional areas, were identified.

- Collaboration Tools - CHEF
- Data repository
- Data viewer
- Data streamer
- Electronic Lab Notebook
- Telepresence
- Data acquisition
- Hybrid experiments
- Simulation repository
- Underlying GRID infrastructure

3 - Estimation of effort spent on each system component

An estimate of effort was generated for each system component based on the allocation of budget to different system integration activities for the first three years of the project. The WBS structured presented in the Project Execution Plan (http://neesgrid.org/documents/NEESgrid_PEP_03_03_2003.pdf) was used for this analysis.

Budget for first 3 years		
WBS		Budget
1	System Components	$ 3,523,000.00
2	Information Services Components	$ 3,120,370.00
3	Community Outreach	$ 1,055,991.00
4	Management	$ 1,851,301.67
		$ 9,550,662.67

The effort estimate included WBS items 1 and 2, but excluded items 1.2 and 2.2.1.

Budget considered in Effort Estimate		
1	System Components	$ 3,523,000.00
2	Information Services Components	$ 3,120,370.00
1.2	Deployment, Operations & Community Support	$ (1,565,000.00)
2.2.1	Prototype Collaborative Environment	$ (36,050.00)
		$ 5,042,320.00

The table below shows the effort estimate for each system component. Certain WBS items, such as 1.1.1-System Requirements Specification, or 1.1.8-Documentation, were spread across several functional areas.

		Effort Estimate		WBS	
				2.2.3	Adaptation of CHEF for Collab. Svcs. on NEESgrid (grid enabling CHEF, Grid FTP portlet)
1	Collaboration Tools – CHEF	17%	$ 838,306.50	2.2.4	Documentation
				2.4.1	Recommend standards for data/metadata models
				2.4.2	Specification for data services architecture
				2.4.3	APIs for data harvesting, mgmt and access
				2.4.4	Curated data repository
2	Data repository	15%	$ 750,000.00	2.4.5	Documentation
				2.2.2	Visualization support for collaboration tools
				2.2.4	Documentation
3	Data viewer	8%	$ 413,163.50	1.1.2	NEESgrid Early Adoption
				1.1.4	Access to Experimental Apparatus and Instruments (NSDS)
				1.1.2	NEESgrid Early Adoption
				1.1.1	System Requirements Specification
4	Data streamer	7%	$ 352,930.00	1.1.8	Documentation
5	Electronic Lab Notebook				
				2.1.1	Remote Teleobservation
				1.1.2	NEESgrid Early Adoption
				2.1.2	Remote Teleoperation
6	Telepresence	17%	$ 867,150.00	2.1.3	Documentation
				1.1.4	Access to Experimental Apparatus and Instruments (Labview)
				1.1.2	NEESgrid Early Adoption
				1.1.1	System Requirements Specification
7	Data acquisition	7%	$ 352,930.00	1.1.8	Documentation
				1.1.7	Collaboration Support (NCTP)
8	Hybrid experiments	3%	$ 168,175.00	1.1.1	System Requirements Specification
				2.3.1	Community Simulation Tool Archive
				2.3.2	Repository of Sample Community Code Results
9	Simulation repository	9%	$ 450,000.00	2.3.3	Usability Enhancements for Community Codes
10	Underlying GRID infrastructure	17%	$ 849,665.00	1.1.3	NEESgrid Security Enhancements
				1.1.1	System Requirements Specification
				1.1.6	Access to data
				1.1.4	Access to Experimental Apparatus and Instruments

				1.1.5	Access to simulation codes
				1.1.7	Collaboration Support
				1.1.8	Documentation
		100%	**$ 5,042,320.00**		

4 - Mapping of system components to user requirements

Finally, system components were mapped to one more user requirements. The party responsible for each system component was then asked to answer two questions about each user requirement:

1. Are you doing work that satisfies this requirement? (Y/N)
2. Give a brief description of deliverable(s) that result from this work.

Requirement Traceability Matrix

Information about the user requirements and about the system components was compiled to create the Requirement Traceability Matrix. A summarized version of the RTM is shown below and contains the following information:

- User requirement (category and description)
- System component that should fulfill the user requirement
- Budget Status (budgeted/ not budgeted)
- Assessment of system integrators as to whether the system component indicated addresses the user requirement
- Brief description of deliverable(s) that result from the work performed by the system integrator to fulfill the user requirement

As of June 2003, the Requirement Traceability Matrix indicated that 61.3% of the user requirements were being addressed by the system integration team and that 18.7% were not being addressed. The remaining 20% needed to be discussed further in order to determine whether the system integration effort was adequately fulfilling those user requirements.

Assessment of User Requirements

June 2003			January 2004		
Y	46	61.3%	Y	57	76%
N	14	18.7%	N	18	24%
Need further discussion	15	20.0%			
Total	75	100%	Total	75	100%

In response to the January 2004 NSF site review team, the 20% of user requirements that had been classified as needing further discussion were reevaluated. As of January 2004, 11 of the 15 requirements that had been in need of further discussion have been reclassified as being in scope and addressed by the system integration team, 3 of the 15 requirements are classified as being out of scope, and one requirement is not under SI control. These reclassified requirements are denoted with asterisks in the Requirements Traceability Tables below.

Requirement Traceability Matrix

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
Collect and Store data	Acquire simulation results	Data acquisition	Labview	budgeted	Y	Probably a custom driver; work in early stages now
Collect and Store data	Acquire numerical data from laboratory experiment (automatically)	Data acquisition	Labview	budgeted	Y	Zombie DAQ program in released code
Collect and Store data	Acquire numerical data from laboratory experiment (manually)	Data acquisition	Labview	budgeted	Y	Example DAQ code in release
Collect and Store data	Acquire data from field investigation	Data acquisition	Labview	budgeted	Y	ADXL and DI-194 drivers and example code, http://www.mcs.anl.gov/neesgrid/adxl.html and http://www.mcs.anl.gov/neesgrid/dataq.html
Collect and Store data	Acquire video/photographs from laboratory experiment	Data acquisition	Labview	budgeted	Y	IEEE-1394 (Firewire) code and results, at http://www.mcs.anl.gov/neesgrid/firewire.html
Collect and Store data	Acquire data from other sources (e.g., historical data or non-NEES data) (manually) for comparison/overlay of data	Data acquisition	Labview	budgeted	Y	Protocol docs, example non-Labview code, ADXL & DI194
Collect and Store data	Store simulation and experiment results	Data repository	Data storage	budgeted	Y	NEES File Management Service to be released in Alpha 2
Collect and Store data	Interface allowing researchers to bring along their own data acquisition system, sensors, or payload experiments	Data acquisition	Labview	budgeted	Y	Protocol documentation and examples, on http://www.mcs.anl.gov/neesgrid/
Collect and Store data	Tool for inter-linking related experimentation sessions	Data acquisition	Labview	budgeted	Y	NTCP and related control system work in progress
Collect and Store data	Tools for characterizing a community/location/ structure/project (as basis for search and comparison)	Data repository	Metadata generation	budgeted	Y	the metadata repository will enable the community to characterize their work using metadata objects

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
Collect and Store data	Easy tool for supplying meta-information as project proceeds (electronic notebook)	Data repository	Metadata generation	budgeted	Y	integration of Nestor's e-notebook with repo (Alpha 2)
Collect and Store data	Metadata ingestion tools (.provide the capability to excerpt information from the electronic notebook and use it to populate the metadata model)	Data repository	Metadata generation	budgeted	Y	Alpha 2 - batch uploading in XML format
Collect and Store data	Manual metadata input (human sensor data recording as a specific metadata acquisition agent)	Data repository	Metadata generation	budgeted	Y	Alpha 1 - CHEF GUI
Collect and Store data	Store metadata	Data repository	Metadata generation	budgeted	Y	NEES Metadata Service - Alpha 1
Collect and Store data	Common/standard data and metadata formats	Data repository	Metadata generation	budgeted	Y	Community will develop with guidance from SI. ongoing. Strawman released. data working in Aug 03
Collect and Store data	Support for multiple data formats	Data repository	Metadata generation	budgeted	Y	framework for format conversion and example implementations will be provided. data repository will support any format for file management capabilities, metadata repository will support XML interchange format and other formats can be translated into that format
Collect and Store data	Develop a metadata registry which enforces business rules for specification of metadata elements	Data repository	Metadata generation	budgeted	Y*	Validation will be supported
Collect and Store data	Metadata-driven lifecycle management for objects	Data repository	Metadata generation	budgeted	N*	Lifecycle management is moot – NEESgrid will have a permanent repository with versioning
Collect and Store data	Utilize controlled vocabulary	Data repository	Metadata generation	budgeted	Y	That is validation - will be supported in final release
Collect and Store data	Data streaming/ automatically archive experimental data to central repository (OR to local archive first and then to central, after validation)	Data streamer	NSDS	budgeted	Y*	

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
Collect and Store data	Optimize data compression while retaining data quality	Data repository	Data storage	budgeted	N	
Search data	Search data and metadata	Data repository	Data discovery	budgeted	Y*	Metadata search will be supported in final release
Search data	Search for raw data and data derivatives	Data repository	Data discovery	budgeted	N	data search will be supported by associating metadata with data objects and searching the metadata. derived data can be linked to raw data with metadata elements
Search data	Search for historical earthquake-related data	Data repository	Data discovery	budgeted	Y	a framework for interface with external data resources, and example implementation for external sources of earthquake-related data (e.g., ANSS) will be provided
Manage data	Common interfaces to widely used tools (e.g. Matlab)	Data viewer	Stored data viewing	budgeted	N	Native Chef WT.NG tool for viewing archived experiment data and common UI to collab tools, but not simulation tools
Manage data	Manipulate experiment data (using tools such as Excel, MathCAD and Mathematica)	Data repository		NOT budgeted	N	
Manage data	"Clean" experiment data	Data repository		NOT budgeted	N	
Database Administration	Remote database management	Data repository		NOT budgeted	Y*	the CHEF interface allows remote management of the repository and the data/metadata in it
Database Administration	Backup-restore/archive-recovery of data	Data repository		NOT budgeted	Y	(at the central repository) -- sites must provide site-specific backup capabilities
Database Administration	Assure data integrity	Data repository		NOT budgeted	Y*	

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
Hybrid experiments	Hybrid experiments	Hybrid experiments	NTCP	budgeted	Y*	NSDS, NTCP, NTCP drivers for proxy server and Matlab, Matlab interfaces to NTCP (being done by Erik Johnson) We provide interfaces to support hybrid experiments. However, sites need to design their own experiments, write their own computational simulation code, and provide (in some cases) drivers to interface with their own local equipment.
Data Viewing	Time synchronization of data and video streams	Data viewer	Real-time data viewing Stored data viewing	budgeted	Y	Options in data viewer for selecting multiple channels of stored data, and displaying the data synchronized with their time stamps.
Data Viewing	Real time access to visualization of sensor data	Data viewer	Real-time data viewing	budgeted	Y	Option on data viewer to display real time experiment data.
Data Viewing	Access to visual, text, and algorithm info from multiple sites during experiment	Data viewer	Real-time data viewing Stored data viewing	budgeted	Y	Providing data viewer which gives access to experiments from multiple sites (but not multiple sites during a single experiment, nor Telepresence)
Data Viewing	Tool for overlaying data	Data viewer	Real-time data viewing Stored data viewing	budgeted	Y	Option in Data Viewer to display multiple channels in one view.
Data Viewing	3D visualization tools for analyzing results	Data viewer	Real-time data viewing Stored data viewing	budgeted	N	
Synchronous collaboration	Teleoperation	Telepresence	Remote Teleoperation	budgeted	Y*	Teleoperations for remote controlled camera is operational. Teleoperations for instruments is now under the direction of the ISI Team which is defining NTCP and will perform all other remote operations. A remote control interface using the reference implementation of Labview has been written by ANL to operate with NTCP.

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
Synchronous collaboration	Teleobservation	Telepresence	Remote Teleobservation	budgeted	Y*	Observation of the experimental laboratory space using Video Cameras is operational and handled by the TPM thin-www client. Streaming data from sensors is now handled by NSDS and displayed by the CHEF Viewer . A NSDS interface using the reference implementation of Labview has been completed.
Synchronous collaboration	Remote control of cameras	Telepresence	Remote Teleobservation	budgeted	Y	This is operational in the TPM thin-www client interface.
Synchronous collaboration	Ability to handle very high-speed video in real time (for telepresence)	Telepresence	Remote Teleobservation	not budgeted	N	The Video system design operates at a maximum rate of NTSC not "very high speed". Very High Speed video is out of scope.
Synchronous collaboration	Chat	Collaboration Tools - CHEF	Chat	budgeted	Y	Native Chat tool being developed as part of WT.NG
Synchronous collaboration	Videoconferencing services (discovery of MCUs)	Underlying GRID infra.	Collaboration services	budgeted	N	Videoconferencing will be handled through off-the-shelf technology (e.g., Polycom)
Synchronous collaboration	Track who else is on	Collaboration Tools - CHEF	Presence awareness	budgeted	Y	User Present list developed as part of standard WT.NG
Synchronous collaboration	Data conferencing (remote sharing of data)	Collaboration Tools - CHEF		NOT budgeted	N	
Synchronous collaboration	Shared whiteboard with telepointers	Collaboration Tools - CHEF		NOT budgeted	N	
Asynchronous collaboration	Document version control	Collaboration Tools - CHEF		NOT budgeted	N	
Asynchronous collaboration	Notify collaborators about changes or additions to shared work	Collaboration Tools - CHEF	Resource sharing	budgeted	Y	Synopsis tools showing recent activity, and notification service to let users know of changes being developed as part of WT.NG

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
Other Collaboration Tools - Synchronous and Asynchronous	Threaded discussion tool	Collaboration Tools - CHEF	Threaded discussion	budgeted	Y	Native Chef tool being developed as part of WT.NG
Other Collaboration Tools - Synchronous and Asynchronous	Video/audio recording and session playback	Data acquisition Data viewer	Labview Stored data viewing	budgeted	Y	**Data acquisition:** IEEE-1394 (Firewire) results and code, http://www.mcs.anl.gov/neesgrid/firewire.html **Data viewer:** We are working on a stored (archived) data viewer, but not on recording audio or video.
Other Collaboration Tools - Synchronous and Asynchronous	Scheduling application (people and resources)	Collaboration Tools - CHEF	Calendaring and scheduling	budgeted	Y	Native Chat tool being developed as part of WT.NG
Other Collaboration Tools - Synchronous and Asynchronous	Individual online workspaces	Collaboration Tools - CHEF	Workspace policies	budgeted	Y	Native Chef capability being developed as part of WT.NG
Other Collaboration Tools - Synchronous and Asynchronous	Share files	Collaboration Tools - CHEF	Resource sharing	budgeted	Y	Native Chat tool for Resources being developed as part of WT.NG
Other Collaboration Tools - Synchronous and Asynchronous	Privacy/Reciprocity	Collaboration Tools - CHEF	Workspace policies	budgeted	Y	Permissions controls being developed as part of Chef, WT.NG
Other Collaboration Tools - Synchronous and Asynchronous	Tools to search for people and their interests (discover/collaborate)	Underlying GRID infra.	Security enhancements/CAS	budgeted	N*	Many options exist to meet this need – therefore, this requirements was never stipulated as within the scope of work for NEESgrid
Simulation codes	Store simulation codes (through Portal)	Simulation repository	Simulation repository	budgeted	Y	Sim tool repository/links (see SA document)
Simulation codes	Access simulation codes (through Portal)	Simulation repository	Simulation repository	budgeted	Y	Capability to run community code remotely
Simulation codes	Run data thru simulation codes	Simulation repository	Simulation repository	budgeted	Y	Analysis content portions of repository

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
Simulation codes	Version control of simulation codes	Simulation repository	Simulation repository	budgeted	Y	outline of version history of tools in repository
Simulation codes	Reference documentation for code repository	Simulation repository	Simulation repository	budgeted	Y	analysis content library showing how sim tools used
Simulation codes	Design portal interface to common codes	Simulation repository	Simulation repository	budgeted	Y	a portal interface to at least one community code
Repository Curation	Data audit protocols	Data repository	Data storage	budgeted	Y*	
Repository Curation	Quality analysis of software (simulation codes)	Simulation repository	Simulation repository	budgeted	N*	With the change in the simulation sub-award on NEESgrid, there is no longer a simulation respository activity.
Access to high performance computing resources	Provide access to high performance computing	Underlying GRID infra.	Resource discovery	budgeted	Y?	Actually, "access" covers three things -- providing physical resources (i.e., supercomputers), providing tools for discovering those resources, and providing tools to schedule those resources. We're not providing the physical resources, but we are are providing tools to discover them (deliverable is MDS) and schedule them (deliverable is GRAM). Both GRAM and MDS are part of the Globus Toolkit.
Access to high performance computing resources	Schedule high performance computing resources	Underlying GRID infra.	Resource discovery	budgeted	Y	GRAM (part of the Globus Toolkit). However, I wouldn't call this "resource discovery"; it would be "resource scheduling" or "resource management".
Access to high performance computing resources	Prioritize allocation of resources	Underlying GRID infra.	Resource discovery	budgeted	N*	HPC resources are allocated as national resources under a separate review process, independent of

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
						NEESgrid and not under the control of the NEESgrid team.
Security	Permission controls	Underlying GRID infra.	Security enhancements/CAS	budgeted	Y	CAS, and all the individual services we're providing (e.g., GRAM, NSDS, NTCP, gridftp, the metadata tools, CHEF).
Security	Security of system and data	Underlying GRID infra.	Security enhancements/CAS	budgeted	Y*	No one can guarantee "security of system and data". We have designed the architecture to make the systems and networks no less secure than they would have been without NEESgrid (e.g., we support -- and encourage -- isolating the DAQ and control systems behind firewalls). Much of the software we are providing (e.g., the components that are part of the Globus Toolkit) have undergone security reviews and are in broad use in very security-conscious environments. We are providing security features, such as GSI for authentication and message integrity.
Security	Single security architecture	Underlying GRID infra.	Security enhancements/CAS	budgeted	Y	GSI (part of the Globus Toolkit)
Safety	Minimum number of local personnel required for remote operations	Telepresence Hybrid experiments	Remote Teleoperation NTCP	not budgeted	N	We support having fewer local personnel than you would have to without NSDS and NTCP. "Minimum number" sounds pretty absolute. We're certainly not doing anything, for example, to reduce the number of people required to build an experiment specimen.

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
Safety	Provide a “kill” switch (emergency shutdown of an experiment)	Telepresence Hybrid experiments	Remote Teleoperation NTCP	not budgeted	N	We provide an interface to NTCP to cancel outstanding operations, but it's up to the sites to install drivers that shut things down safely. We also cannot, since NTCP messages travel over the commodity Internet, guarantee how quickly a cancel message will be received. Sites should have local personnel supervising their experiments and should implement their own local "kill switches" for safety.
Safety	Enforce guidelines to operate within equipment tolerances	Telepresence Hybrid experiments	Remote Teleoperation NTCP	not budgeted	N	NTCP includes a pluggable interface that supports this, but it's up to the individual sites to install drivers that make the appropriate checks. It's also up to individual sites to set up appropriate (non-software) policies and procedures to protect their equipment and personnel.
Usability	Common GUI integrating data, simulation, video and visualization (standard interfaces)	Collaboration Tools - CHEF		budgeted	Y	Style guide to promote and CSS to force tool GUI consistency
Usability	Ease of use	Collaboration Tools - CHEF		budgeted	Y	Tools developed with a common GUI across tools in a web browser environment.
Usability	Reusable layout configuration	Collaboration Tools - CHEF		budgeted	Y	CSS being used to provide skinning options.
Usability	Platform independence	Collaboration Tools - CHEF		budgeted	Y	Chef, WT.NG being developed to require a browser only, no additional plugins.

The table below shows the user requirements that are not being addressed by the system development effort. They correspond to 18.7% of the user requirements.

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
Collect and Store data	Optimize data compression while retaining data quality	Data repository	Data storage	budgeted	N	
Search data	Search for raw data and data derivatives	Data repository	Data discovery	budgeted	N	data search will be supported by associating metadata with data objects and searching the metadata. derived data can be linked to raw data with metadata elements
Manage data	Common interfaces to widely used tools (e.g. Matlab)	Data viewer	Stored data viewing	budgeted	N	Native Chef WT.NG tool for viewing archived experiment data and common UI to collab tools, but not simulation tools
Manage data	Manipulate experiment data (using tools such as Excel, MathCAD and Mathematica)	Data repository		NOT budgeted	N	
Manage data	"Clean" experiment data	Data repository		NOT budgeted	N	
Data Viewing	3D visualization tools for analyzing results	Data viewer	Real-time data viewing Stored data viewing	budgeted	N	
Synchronous collaboration	Ability to handle very high-speed video in real time (for telepresence)	Telepresence	Remote Teleobservation	not budgeted	N	The Video system design operates at a maximum rate of NTSC not "very high speed". Very High Speed video is out of scope.
Synchronous collaboration	Videoconferencing services (discovery of MCUs)	Underlying GRID infra.	Collaboration services	budgeted	N	It's not clear that this is still a requirement; however, if it is, the deliverable would be an MDS provider listing MCU resources
Synchronous collaboration	Data conferencing (remote sharing of data)	Collaboration Tools - CHEF		NOT budgeted	N	
Synchronous collaboration	Shared whiteboard with telepointers	Collaboration Tools - CHEF		NOT budgeted	N	

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
Asynchronous collaboration	Document version control	Collaboration Tools - CHEF		NOT budgeted	N	
Safety	Minimum number of local personnel required for remote operations	Telepresence Hybrid experiments	Remote Teleoperation NTCP	not budgeted	N	We support having fewer local personnel than you would have to without NSDS and NTCP. "Minimum number" sounds pretty absolute. We're certainly not doing anything, for example, to reduce the number of people required to build an experiment specimen.
Safety	Provide a “kill” switch (emergency shutdown of an experiment)	Telepresence Hybrid experiments	Remote Teleoperation NTCP	not budgeted	N	We provide an interface to NTCP to cancel outstanding operations, but it's up to the sites to install drivers that shut things down safely. We also cannot, since NTCP messages travel over the commodity Internet, guarantee how quickly a cancel message will be received. Sites should have local personnel supervising their experiments and should implement their own local "kill switches" for safety.
Safety	Enforce guidelines to operate within equipment tolerances	Telepresence Hybrid experiments	Remote Teleoperation NTCP	not budgeted	N	NTCP includes a pluggable interface that supports this, but it's up to the individual sites to install drivers that make the appropriate checks. It's also up to individual sites to set up appropriate (non-software) policies and procedures to protect their equipment and personnel.

Included in the table below are the 20% of user requirements that required further discussion in June 2003, and their classification as of January 2004.

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
Collect and Store data	Develop a metadata registry which enforces business rules for specification of metadata elements	Data repository	Metadata generation	budgeted	Y*	I don't know what "business rules" means. validation will be supported in Alpha 2
Collect and Store data	Metadata-driven lifecycle management for objects	Data repository	Metadata generation	budgeted	N*	I don't know what "lifecycle management" means. The repo supports versioning in Alpha 1
Collect and Store data	Data streaming/ automatically archive experimental data to central repository (OR to local archive first and then to central, after validation)	Data streamer	NSDS	budgeted	Y*	
Database Administration	Remote database management	Data repository		NOT budgeted	Y*	what is meant by "database"? the CHEF interface released in Alpha 1 allows remote management of the repository and the data/metadata in it
Database Administration	Assure data integrity	Data repository		NOT budgeted	Y*	what kind of integrity?
Other Collaboration Tools - Synchronous and Asynchronous	Tools to search for people and their interests (discover/collaborate)	Underlying GRID infra.	Security enhancements/CAS	budgeted	N*	The description does not match the "system function". Searching for people with common interests is not a security enhancement, and this is not a function of CAS. If this is a requirement, the solution would probably be to deploy LDAP servers using the Internet2 schemas for describing people; most of that work would probably be done by the deployment team.
Repository Curation	Data audit protocols	Data repository	Data storage	budgeted	Y*	what kind of protocols?

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
Access to high performance computing resources	Prioritize allocation of resources	Underlying GRID infra.	Resource discovery	budgeted	N*	This is a pretty open-ended requirement. Calling it "resource discovery" makes it sound like the requirement is to provide allocation committees information about resources to aid them in setting priorities -- in this case, the deliverable is MDS. If the requirement is to enforce decisions about what priorities have been decided, then GRAM supports whatever underlying mechanism exists at the computing site. Also, since this is under "access to high performance computing resources", I assume this requirement pertains to high performance computers. For other resources and services, there are mechanisms that allow administrators to specify who can do what, and when (deliverables would be CAS plus many individual services provided as part of NEESgrid).
Security	Security of system and data	Underlying GRID infra.	Security enhancements/CAS	budgeted	Y*	No one can guarantee "security of system and data". We have designed the architecture to make the systems and networks no less secure than they would have been without NEESgrid (e.g., we support -- and encourage -- isolating the DAQ and control systems behind

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
						firewalls). Much of the software we are providing (e.g., the components that are part of the Globus Toolkit) have undergone security reviews and are in broad use in very security-conscious environments. We are providing security features, such as GSI for authentication and message integrity.
Synchronous collaboration	Teleoperation	Telepresence	Remote Teleoperation	budgeted	Y*	Teleoperations for remote controlled camera is operational. Teleoperations for instruments is now under the direction of the ISI Team which is defining NTCP and will perform all other remote operations. A remote control interface using the reference implementation of Labview will be written by ANL but it is waiting for NTCP to be completed by ISI.
Synchronous collaboration	Teleobservation	Telepresence	Remote Teleobservation	budgeted	Y*	Observation of the experimental laboratory space using Video Cameras is operational and handled by the TPM thin-www client. Streaming data from sensors is now handled by NSDS and displayed by the CHEF Viewer . A NSDS interface using the reference implementation of Labview has been completed.
Search data	Search data and metadata	Data repository	Data discovery	budgeted	Y*	Metadata search will be supported in final release, but search of data will only be supported in a limited fashion

User Requirement		System Component		Budget Status	Are you doing work that satisfies this requirement? (Y/N)	Brief description of deliverable(s) that result from this work
Hybrid experiments	Hybrid experiments	Hybrid experiments	NTCP	budgeted	Y*	NSDS, NTCP, NTCP drivers for proxy server and Matlab, Matlab interfaces to NTCP (being done by Erik Johnson) We provide interfaces to support hybrid experiments. However, sites need to design their own experiments, write their own computational simulation code, and provide (in some cases) drivers to interface with their own local equipment.
Access to high performance computing resources	Provide access to high performance computing	Underlying GRID infra.	Resource discovery	budgeted	Y?	Actually, "access" covers three things -- providing physical resources (i.e., supercomputers), providing tools for discovering those resources, and providing tools to schedule those resources. We're not providing the physical resources, but we are are providing tools to discover them (deliverable is MDS) and schedule them (deliverable is GRAM). Both GRAM and MDS are part of the Globus Toolkit.
Repository Curation	Quality analysis of software (simulation codes)	Simulation repository	Simulation repository	budgeted	N*	repository metadata that will include QA metrics

Conclusion

The Requirements Traceability Matrix developed for the NEESgrid project indicates that 76% of the user requirements are being addressed by the system integration team and will be implemented in the first version of the system. 24% of the user requirements are not being addressed by the system integration effort. These numbers do not represent the effort spent on user requirements relative to each other. Each user requirement was considered as one unit, independently of the amount of system integration effort required for each one.

The generation of the Requirements Traceability Matrix was based on project documentation and on conversations with the System Integration team.
It required approximately 140 man-hours to be completed. The user requirements included in the matrix represent the points of view of several constituencies involved in the NEESgrid project. It accounts for the requirements documented by NSF in its Program Solicitation. It also accounts for the requirements gathered from users in many occasions, including the Workshop held on October 23-25, 2002 at the Timberline Lodge, Mt. Hood, OR. The matrix includes the perspective of the system integration team as summarized in the System Overview and in the System Architecture documents. In addition, the RTM accounts for the viewpoint of the site review teams as reflected in the site visit reports from March 2001, 2002 and 2003 and January 2004. The Requirements Traceability Matrix is a rich compilation of all of the user requirements surfaced by the NEESgrid project to this date.