7 Grid Computing and the Future of Neuroscience Computation

John D. Van Horn, James Dobson, Michael Wilde, Jeffrey Woodward, Yong Zhao, Jens Voeckler, and Ian Foster

Since the introduction of computing as a vital tool in scientific research, we have seen several distinct generations of systems architectures come and go. With the earlier generations, it was uncommon for individual researchers to have computers on their desks, let alone computers dedicated solely to their particular research purposes. Mainframe, vector, and time-shared VMS or UNIX computers served as shared resources supporting the concurrent analyses of many researchers. Today, of course, computers are ubiquitous in many parts of the scientific world; researchers typically have one or more personal computers at their disposal. The days of large, monolithic systems are over, and individual computers are the rule. Recently, however, a new computing paradigm has emerged that will again alter the way data computations are performed. Innovative infrastructure technologies including high-speed networking, inexpensive hardware solutions, and open-source, standards-based software have brought a paradigm shift in how computational research is performed. Using local high-density central processing units (CPUs) as well as geographically distributed systems, the high-throughput processing of large quantities of neuroimaging data, in particular, can be completed on the order of minutes rather than hours. This is a fundamental concept behind Grid computing and is causing considerable excitement in the scientific community (Foster, 2003; Foster and Kesselman, 1998).

Large-scale computing has clear advantages for the analysis of many types of large scientific data sets—from particle physics to gene expression, from geospace to astrophysics. For the neurosciences, Grid computing has direct applicability for the analyses of the terabytes of neuroimaging data contained in public archives such as the fMRI Data Center (fMRIDC; see http://www.fmridc.org), based at Dartmouth College (Van Horn et al., 2001). This publicly available repository includes complete data sets from published studies of human cognition using functional magnetic resonance

imaging (fMRI). Data sets include four-dimensional functional image volume time course data (e.g., EPI, Spiral), high-resolution brain anatomy–imaging data (e.g., SPGR, MPRAGE), study metadata, and supporting data collected as part of the study. The fMRIDC curates, anonymizes, and packages these data sets for open dissemination; researchers around the world may then use the data sets to conduct novel analyses, test alternative hypotheses, explore new means of data visualization, or use them in education and training. But this ever increasing collection of study data sets can also be compared en masse across studies to identify patterns of cognitively induced signal change unseen in any individual study. Such comparative studies will enable neuroscientists to better understand how particular combinations of brain regions contribute to the processing of cognitive information across different individual task paradigms, and where those paradigms share particular cognitive components.

This chapter discusses the concepts of distributed high-performance and high-throughput Grid computing systems and presents specific examples of their application in the large-scale processing of functional neuroimaging data. Featuring data sets from the fMRI Data Center, we demonstrate the improvement in throughput that may be obtained in (1) using highly parallel clustered processing systems; and (2) utilizing geographically distributed cluster computing power for the analysis of brain-imaging data. We describe a promising opportunity to leverage the capabilities of two rapidly evolving and closely paradigms in information technology, Grid computing and virtual data, to provide the tools, techniques, and resources to make the fMRIDC archive an indispensable community resource for driving advances in neuroscience research. We also describe how Grid computing can harness multiple distributed computing facilities to solve large problems, and how the Virtual Data Model of computing can help scientists utilize the Grid environment with mechanisms for expressing workflows in a high-level language that frees them from distributed computing concerns and at the same time creates accurate, sharable, re-executable descriptions of how every step in a complex, long-running data analysis effort was carried out.

More specifically, we explain how highly optimized anatomical brain-warping processing pipelines can be made to run on the order of minutes rather than hours, and how complex multistage analyses of terabytes—thousands of gigabytes—of data can be distributed across multiple processors and systems. These processes do not require specialized hardware and can often be controlled from laptop computers using Web-enabled interfaces. We also describe how "virtual data–based" workflow systems can

serve as an enabling interface to the Grid to facilitate reproducible data analyses and knowledge-intensive collaboration.

As neuroscience researchers make increasing use of functional neuroimaging for the mapping of cognitive activity, they will come to rely more and more on Grid computing. Our chapter provides an overview of what we believe to be a critical aspect in the future of functional neuroimaging data analysis.

Development

Although the advent and development of MRI technology over the past two decades have resulted in a quantum leap in our ability to measure the brain's capabilities, this technology has also vastly increased the amount of data brain researchers must manipulate, manage, and store. In contrast to those typical for positron-emission tomography (PET) studies, data sets for functional MRI studies are large, often exceeding several gigabytes (GB) in size. As advances are made in MRI scanner technology to permit the more rapid acquisition of data, functional imaging experiments will consist of ever greater amounts of data per unit of time over the same scan duration. And as cognitive neuroscientists ask ever more sophisticated questions about fundamental brain processes, they will undoubtedly collect data on an ever greater number of subjects and fMRI time courses per subject.

Data Archiving

Indeed, within the next decade, neuroimaging data archives containing several petabytes—several million gigabytes—are not out of the question and will likely become the norm. A number of individual fMRI data sets already rival the full size of many extant large genetic (Benson et al., 2003) and protein (Berman et al., 2002) science data archives. For instance, the complete study data set from Buckner et al., 2000, exceeds 20 GB. It can be expected that, as advances are made in the spatiotemporal resolution of MRI scanners, the amount of published brain image data will routinely rival the amount contained in the human genome database. A challenge therefore exists in devising efficient means for comparing and contrasting these data on a large scale but within reasonable time.

With these issues in mind, the fMRI Data Center was established as a public archive for fMRI study data and the associated experimental metadata (Van Horn et al., 2004). The fMRIDC began receiving data from researchers in 2000 and made data sets publicly available in 2001. At

present, the archive contains more than 100 complete data sets, which researchers may request online and have shipped to them free of charge. Authors of fMRI studies are asked to provide the details of their experiments across several levels, including scanner imaging protocols, subject demographics, and experimental design. The principal intent of obtaining such complete information about each study is to permit researchers to reconstruct the results reported in the literature by the original authors.

The average fMRIDC data set contains approximately 18 gigabytes of uncompressed data and may contain over 30,000 files. A typical functional run is composed of a time series of three-dimensional ANALYZE-compatible volumes, where each ANALYZE volume/header pair represents a single point of time in the time series. A functional run may have 200 files or more where the volumetric data are around 1 megabyte per volume. Recent studies have grown much larger: the fMRIDC holds one anatomical study with over 300 subjects, and a recent functional study comprising over 100 subjects. This scale of data represents a tremendous opportunity for the sharing, and growth, of knowledge in the field. It also presents a significant challenge: to make this information both available to and usable by the neuroscience research community, we must first solve many problems in data transport and storage and in making available and usable the computational tools and resources to make further use of the archived study data.

Grid Computing

As archives of neuroscience data expand, the computational needs required to process the immense volume of data also increase. In terms of the memory and processor speed requirements, individual desktop workstations are now no longer suitable for analyzing potentially gigabytes worth of data at a time. What is needed is to combine effort from across multiple, distributed processing elements and take advantage of the collective power they possess toward analyses of these neuroscientific resources on a massive scale.

The Grid is an emerging paradigm in the computer science community based around distributed systems of computers, data, tools, and people (Foster, 2003). The term *Grid* was coined in the mid-1990s to denote a proposed distributed computing infrastructure for advanced science and engineering. The specific challenge that motivates interest in Grid computing is the coordination of resource sharing and problem solving in dynamic,

multi-institutional, spatially dispersed, virtual organizations. Strictly speaking, this does not refer to file exchange, but rather to direct access to computers, software, data, and other resources, as required by a range of collaborative problem-solving and resource-brokering strategies emerging in industry, science, and engineering. This sharing is, necessarily, highly controlled, with resource providers and consumers defining clearly, carefully, and exactly what is shared, who is allowed to share, and the conditions under which sharing occurs. The related term *data Grid* was coined to denote a Grid system that has an additional strong emphasis on the sharing of large amounts of data and data storage resources.

The internal organization or fabric of the Grid is evolving with standards and practices emerging from organizations such as the Global Grid Forum (GGF; see http://www.ggf.org). Although the Grid is a framework for collaboration between virtual organizations on a large, geographically diverse, scale, we should note that the Grid need not have a centralized infrastructure and makes minimal assumptions of computing systems homogeneity. Its underlying architecture is being specifically designed for distributed authentication, authorization, storage, and computation. These guidelines and software protocols will enable technologies from multiple research projects and vendors to interoperate on the shared Grid infrastructure.

Most fMRI data are processed using a series of computational algorithms that "pipeline"—apply spatial realignment, slice-timing adjustment, spatial normalization, filtering, and other such operations on—the data. Pipelined processing operations are designed to take one or more fMRI time series and prepare the image volume data for later statistical processing. These multistage pipelines can typically be run on the Grid with the simple inclusion of some basic code infrastructure for locating data and understanding job dependencies. Thus data-intensive sciences such as functional neuroimaging can utilize Grids for analysis of data sets on scales not possible using local computation.

As researchers' needs exceed the computing power available in a desktop environment, they typically turn toward shared departmental or collaboration-run systems ranging from the departmental servers of a few years ago, to supercomputers, to the computing clusters of today. Although cluster computing at various scales has rapidly replaced both departmental servers and supercomputers, the concept of the Grid has spanned all of these changes. Originally termed *metacomputing* (Catlett and Smarr, 1992), the Grid first consisted of geographically separated supercomputers linked to work on single problems in a loosely coordinated fashion, yet its principles

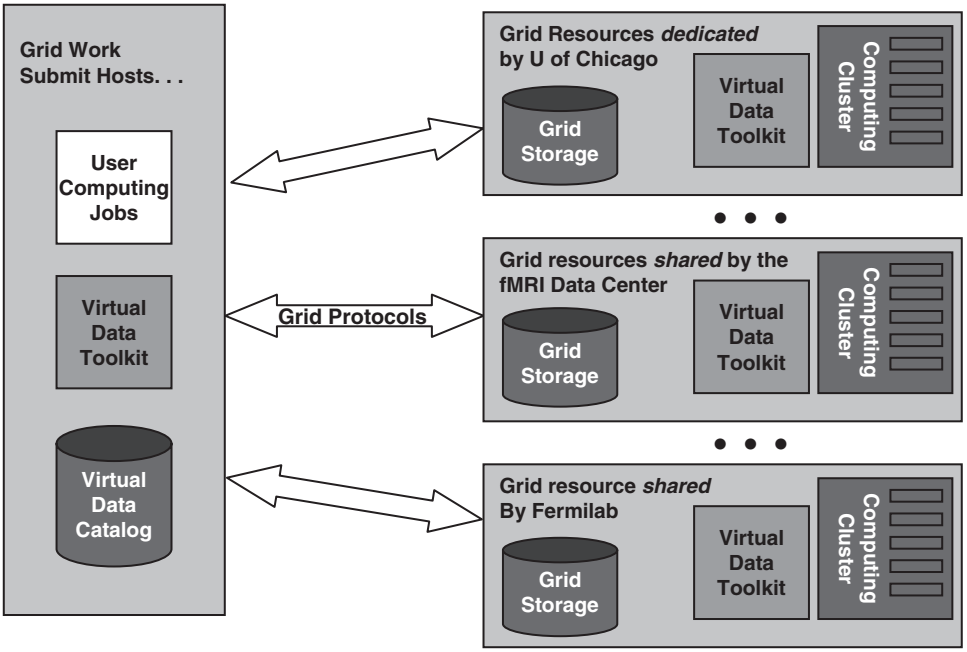
remain much the same even as it links distributed computing clusters to form a single, large "cluster of clusters."

A typical Grid cluster consists of a set of distinct computers, typically similar or identical, called "worker nodes," which provide the computing power of the cluster. A "head node" computer runs a software application called a "scheduler," which distributes computing jobs to the worker nodes. The scheduler maintains a "job queue" of work for the worker nodes. Typically the running of a program or script, a job involves sending the program and input files to a worker node, running the program, and sending the resulting output files back to the user. Some jobs may use a single node to run a program, whereas others, using parallel programming tools such as the "message-passing interface" (MPI) may harness a set of nodes running concurrently and exchanging messages to coordinate and perform parallel computations. A file server provides a file system that is typically shared among all worker nodes and the head node. Files going to and from jobs may reside on this shared file system, or on private file systems that reside on each worker node. Finally, a network such as commodity Ethernet or the higher-speed Myrinet (see http://www.myri.com) is required to link these computers together. For researchers using a Grid cluster, the Grid appears as modeled below in figure 7.1.

Grids and their user communities are often organized as "virtual organizations" (VOs), which consist of groups of collaborating users, typically from multiple real-world organizations such as university departments and laboratory divisions or corporate departments, that are working together to solve a problem or related set of problems (Foster, Kesselman, and Tuecke, 2001). The member organizations that make up a VO may also contribute resources for community use, such as allocation of computing clusters and network-accessible storage, or in some cases, a physical instrument of significant scale and community value such as an electron microscope, a particle accelerator, a fusion reactor, or an earthquake engineering "shake table." Virtual organizations may have no physical resources of their own, but may be granted allocations of resources

Figure 7.1
(*a*) Researcher's view of the Grid. Users submit compute jobs using the virtual data language and drawing from a virtual data catalog. Resources from distributed sites are dedicated (e.g., fully supporting Grid-related activity) or shared (e.g., local jobs given priority) and supply contributed Grid services toward job completion. Note that Grids may vary in size to include the computers in a single laboratory, a building, a campuswide system, or an entire country. (*b*) Grid3 currently consists of 30+ sites in the United States and South Korea.

(a)


Grid Work
Submit Hosts. . .
User
Computing
Jobs
Virtual
Data
Toolkit
Virtual
Data
Catalog
Grid Protocols
Grid Resources dedicated
by U of Chicago
Grid
Storage
Virtual
Data
Toolkit
Computing
Cluster
Grid resources shared by the
fMRI Data Center
Grid
Storage
Virtual
Data
Toolkit
Computing
Cluster
Grid resource shared
By Fermilab
Grid
Storage
Virtual
Data
Toolkit
Computing
Cluster

(b)


UWMilwaukee
UWMadison
FNAL_CMS
FNAL_CMS2
FNAL_CMS3
UM_ATLAS
UC_ATLAS_Tier2
ANL_Jazz
Purdue-Physics
IU_ATLAS_Tier2
PSUIG-id3
Hopkins
HU_mantles
BU_ATLAS_Tier2
BU_AGT_Ti
BNL_ATLAS
UBuffalo-CCR
PDSF
Caltech-PG
CalTech-Grid3
UCSanDiego
UCSanDiegoProd
UNM_HPC
OU_OSCER
OUHEP
Vanderbilt
UTA-DPCC
SMU_ATLAS
Rice-Grid3
UFlorida-PG
UFlorida-Grid3
CHEPREO_FIU
KNU
South Korea
Sun Nov 14 14:26:45 EST 2004

contributed by other communities or funding agencies. Several large-scale Grid VO projects are worth noting for their scale as well as their broad-based, research-oriented resource sharing.

TeraGrid
A virtual computing infrastructure funded by the National Science Foundation (NSF), the TeraGrid is built from five (ultimately, nine or more) individual clusters (themselves built from smaller computers, or nodes), having nearly 1 petabyte of networked disk storage and linked by a *cross-continent* network backbone with a transmission speed of 40 gigabytes per second—four times faster than the fastest *local* area commodity networks currently in use. Its name reflects the scale of the computing power that it provides: 20 teraflops—a trillion floating point operations per second.

Grid Physics Network
A large-scale project also funded by the NSF, GriPhyN was created to explore ways to harness Grid computing to enhance the scientific productivity of large-scale, collaborative, data-intensive physics experiments. GriPhyN applies the concept of "virtual data" to make scientific data-processing workflows and the utilization of distributed resources as easy for individuals and collaborations to use as a single, local computing system.

International Virtual Data Grid Laboratory, Particle Physics Data Grid, and Grid3
Two other large-scale users of the Grid, IVDGL and PPDG have worked in close collaboration with GriPhyN to create Grid3—a large-scale international test Grid of unprecidented scale, pooling the resources of over 30 sites to form an aggregate computing resource with over 3500 CPUs and many terabytes of storage (figure 7.1b). Examples of several such Grid3 projects currently under way are listed in table 7.1.

European Data Grid, Large Hadron Collider Grid, and Enabling Grids for E-Science in Europe
A project sponsored by the European Commission to develop tools for data-intensive science, the European Data Grid (EDG) was created to serve the needs of the high-energy physics community in analyzing data acquired from the Large Hadron Collider (LHC), based at CERN in Geneva. The EDG evolved in 2004 into a new project, Enabling Grids for E-Science in Europe (EGEE; see http://public.eu-egee.org), which combines earlier high-energy physics with biomedical applications.

Table 7.1
Several leading Grid3 projects for high-volume, high-throughput distributed computing

Project Name	Description and URL
ATLAS	"A Torroidal LHC Apparatus"—a high-energy physics experiment at the Large Hadron Collider (LHC) at CERN http://atlas.web.cern.ch/Atlas/Welcome.html
BTeV	"B-Physics at the Tevatron"—a high-energy physics experiment being conducted at Fermilab to explore so-called B-Physics. http://www-btev.fnal.gov
CMS	"Compact Muon Solenoid"—a high-energy physics experiment at the Large Hadron Collider (LHC) at CERN http://cmsinfo.cern.ch/Welcome.html
LIGO	The Laser Interferometer Gravitational Wave Observatory—a coalition of research centers for analyzing gravitational wave data. http://www.ligo.org
SDSS	The Sloan Digital Sky Survey—a project to map and analyze all celestial objects in a given region of the sky. http://www.sdss.org
GADU	Genomics Analysis and Database Update http://compbio.mcs.anl/gov

Each Grid site or resource provider brings a set of services, typically, storage elements (SEs) and compute elements (CEs). For instance, there are currently 30 sites on Grid3, each contributing computational and data resources, with more expected soon. Such projects demonstrate the utility of the Grid for bringing together consortia of researchers all interested in drawing from the shared pool of research data and computing resources they all have to offer. Community-oriented fMRI research collaboratives, individually possessing large data sets (e.g., BIRN), or distributed researchers seeking to combine effort in drawing from large-scale data repositories (e.g., the fMRIDC), are prime candidates for neuroscience-oriented Grid VOs.

Principles

Components of the Grid

The Grid requires a rigorously designed collection of protocols and standards by which data and services may be utilized. Many different technologies exist to provide Grid users with catalogs that describe what

resources are present at the different sites of a Grid. These include the Monitoring and Discovery Service (MDS) of the Globus Toolkit, as well other Grid-specific catalogs (see http://www.ivdgl.org/grid2003). Here we describe the principal software components of Grid computing, from which the Grids described above are constructed.

Globus Toolkit
Providing a large set of services for the creation of Grid systems and is produced by the Globus Alliance (see http://www.globus.org), the Globus Toolkit (Foster and Kesselman, 1998) has gone through several major releases, each of which have brought new standards and interfaces for interaction with distributed resources.

Public Key Infrastructure
Grid security is based on the Public Key Infrastructure (PKI), and uses methods of public/private key pairs to give Grid users a "single sign-on" capability: authenticate once, and create a long-lived "proxy" certificate that can be presented to gain access to Grid services such as the ability to run a job on a remote cluster or to transfer files between clusters. A service from the European Data Grid called the "Virtual Organization Membership Service" (VOMS) is used to map identities from the individual VOs to a local UNIX account.

Grid Resource Allocation and Management
The fundamental component for running computer jobs on the Grid is an element of the Globus toolkit called "Grid Resource Allocation and Management" (GRAM). A site's resources may be managed by schedulers such as Condor, the Portable Batch System (PBS), the Load-Sharing Facility (LSF), and many others. GRAM provides Grid users with a single, uniform interface that permits them to execute and manage jobs on any site of a Grid, without knowing the details of the specific cluster that schedules the software managing the resources of those sites. GRAM provides mechanisms for transporting input data to remotely running jobs and for retrieving output back from them; it is supplemented by Condor-G (Frey et al., 2002), which provides the user with a local queue of jobs being sent into the Grid, the ability to manage the relative priorities and submission rates of jobs in that queue, and several other important features.

Grid File Transfer Protocol
A file transfer service based on the Internet RFC standard file transfer protocol, the GridFTP has extensions to support high-volume data traffic. The

Globus GridFTP server supports the Grid Security Infrastructure (GSI) system to enable single sign-on access to data from any site on the Grid. Many large "data challenge" tests have been conducted using GridFTP to transfer at a rate of many gigabytes per second. Indeed, the entire fMRIDC multiterabyte data archive is accessible via GridFTP. This service is used by remote jobs to transfer data to the site where the computation is performed. GridFTP contains many capabilities that provide the high-performance data transfer required of large-scale data-intensive scientific and engineering applications, including sending multiple streams of data between two servers, strapping multiple servers together, and being able to restart a transfer from any point within a large file. GridFTP serves, for batch jobs running across the Grid, as a ubiquitous data transfer protocol, capable of moving data from any Grid site to any other.

Replica Location Service

The strategy of caching or *replication* is supported in Grid by a cataloging system called the "Replica Location Service" (RLS) component of the Globus Toolkit. This service uses a relational database to create local "replica catalogs," which describe what files exist on the site, and index services, which help speed up lookups for a naming service that can scale to tens or hundreds of millions of file entries Grid-wide. The RLS can also be shared by a distributed group of scientists to specify a common input data set and a name space created for the individual output files from experimentation.

Virtual Data Model

Having outlined the basic groundwork by describing both the mission and nature of the fMRIDC and the Grid computing paradigm and tool sets, we now move on to describe first, how the Virtual Data Model serves to make Grid computing easier and scientific data management more conducive to active research, and then (in the "Applications" section) how this concept has been applied to explore the benefits of Grid computing to fMRI research and data management.

Virtual Data Toolkit

Underlying software technology is needed to deliver data to different groups collaborating across sites, to keep the various sites involved in joint projects in sync, and to serve as a technology transfer vehicle to still other collaborations. The Virtual Data Toolkit (VDT) provides just such software technology.

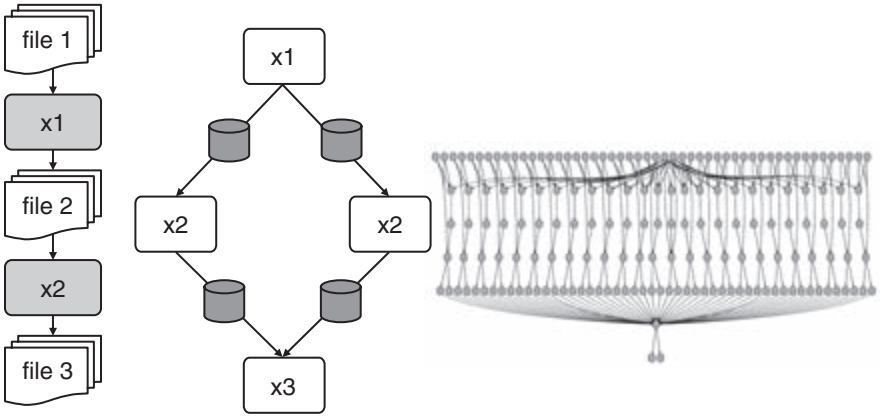
Virtual Data Language

"Virtual data" is a concept that builds on, automates, and seeks to supplement and eventually to replace the traditional practice of recording the steps of a data analysis effort in a logbook or in ad hoc online records. The process of "virtualizing" data involves describing the steps needed (or used) to produce a data object, using a methodical, high-level, declarative specification. Several important benefits derive from this approach, and serve as a productivity lever in the scientific data analysis process. They enable a data object to be produced anywhere in a distributed environment such as the Grid, while freeing the scientist from implementing the complex mechanics of such "location-independent computing." Moreover, they serve to keep accurate, high-fidelity records of how each data object was produced and how data analysis routines and transformation tools were used. They enable intermediate or even "final" data to be discarded and recreated later if necessary (hence making such data "virtual"). The tool set, which enables the use of the virtual data paradigm, is called the "Virtual Data System" (VDS; see http://griphyn. org/vds) and is bundled into the Virtual Data Toolkit to facilitate both the use and the construction of Grid computing environments.

At the heart of the Virtual Data Model is the virtual data language (VDL; Foster et al., 2002) used to express computational procedures for data derivation. The VDL consists of elements to process declarations of workflows and a virtual data catalog in which to store VDL declarations and records of workflow execution. "Planners" map VDL-specified workflows into a sequence of concrete actions for a specific computing environment such as a local system or a distributed Grid. With VDL, the individual computation actions that one defines are typically the invocation of a single application program, query, or Web service. These "atomic" actions are then linked into workflows of arbitrary complexity using VDL statements to declare data derivation steps and functional compositions of these steps. Depending on the level at which the Virtual Data System tool set is used, scientists may either work directly with VDL, or with tools that produce and execute VDL workflows for them.

Most scientific analysis processes to which the Virtual Data Model would be applied involve a "workflow": a series of interdependent steps that lead from input data through a series of data transformations, reductions, filtrations, and so on, to the data sets needed to support and present scientific findings and conclusions. A straightforward workflow typically resembles a "pipeline"—that is, a series of sequential steps in data processing. More complex workflows are described in terms of graphs (from

a)



b)

```
TR tr1(in a1, out a2) {
    argument stdin = ${a1};
    argument stdout = ${a2}; }
TR tr2(in a1, out a2) {
    argument stdin = ${a1};
    argument stdout = ${a2}; }
DV x1->tr1(a1=@{in:file1}, a2=@{out:file2});
DV x2->tr2(a1=@{in:file2}, a2=@{out:file3});
```

Figure 7.2
(*a*) Workflow patterns are routinely visualized in the form of directed acyclic graphs (DAGs), shown here of increasing complexity from left to right. (*b*) Virtual data language (VDL) pseudo-code representation of simple data-processing pipeline DAG.

discrete mathematics) that link a set of data objects with a set of processing steps in arbitrarily complex patterns. Workflows that proceed from beginning to end, with no loops or backtracking ("cycles"), can be expressed as "directed acyclic graphs" (DAGs; figure 7.2). Note that DAGs can be used to indicate both data and control flow dependencies, and, of particular use in distributed computing, to determine which steps of a workflow can be executed in parallel. This is presented in the leftmost DAG in figure 7.2a, where both transformations "× 2" in the diamond can be executed in parallel (figure 7.2b), as can each level of transformation tasks in the middle and rightmost multinode DAGs.

Virtual data language provides two basic declaration statements: transformations and derivations. Transformations encapsulate or "wrap" an application program by providing a specification of the input files read by the application, the output files produced by it, and the arguments it accepts and requires. Transformations are like function declarations in that

they specify a template for invoking an application. Declaring the inputs and outputs of the application is the key to enabling applications to run transparently anywhere in the Grid. Derivations provide the argument values necessary to invoke a transformation. Like function calls in a programming language, derivations can executed immediately, or can be stored for later, possibly repetitive execution. A sample of virtual data language is shown in figure 7.3. The VDL transformation (TR) statement defines a transformation, and the derivation (DV) statement, a derivation. (Note that the TR in a VDL program has no connection at all with the MRI concept of "repetition time"; VDL is an independent, application-neutral language.)

Virtual Data Catalog

This provides a database called a "virtual data catalog" (VDC) for storing transformation and derivation definitions and the invocation records that are created each time a derivation is executed. The VDC serves as a central focal point of scientific data analysis, collaboration, and information recording and sharing. It plays the role of a specialized metadata catalog that contains information about the interfaces of application programs (arguments, input files, and output files), the "recipes" for calling these

```
TR air::alignlinear( in standard_hdr, in standard_img,
                     in reslice_hdr, in reslice_img,
                     out airfile, none args) {
   argument = ${standard_hdr};
   argument = ${reslice_hdr};
   argument = ${airfile};
   argument = args;
}
DV fmridc::2-2004-1234JD-01_run1->air::alignlinear(
    standard_hdr = @{in:"100-3_anonymized.hdr"},
    standard_img = @{in:"100-3_anonymized.img"},
    reslice_hdr  = @{in:"100-5_anonymized.hdr"},
    reslice_img  = @{in:"100-5_anonymized.img"},
        airfile  = @{out:"100-5_anonymized.air"},
           args  =  "-m 6 -t1 80 -t2 80 -s 81 27 3 -q"
);
```

Figure 7.3

Virtual data language derivation to invoke the declarations of an automated image registration (AIR) transformation. The transformation (TR; not to be confused with repetition time in MRI imaging) defines the operation to be performed and the derivation (DV) actually carries out that operation. Derivations may be strung together in workflow scripts to form more complex data-processing pipelines and build up a catalog of commonly used processing operations (e.g., spatial realignment).

applications to perform specific tasks and for deriving data files, logs records on how, when, and where the applications were invoked, how they performed, and how specific data files were produced. The VDC can be queried to provide this information, and such knowledge can play a key role in the process of organizing, specifying, documenting, assessing, or reproducing a data analysis effort.

To begin, researchers require a virtual data catalog, typically in the form of a database hosted by a PostgreSQL, MySQL, or eXist server. They also require a client workstation on which the Virtual Data System is installed and a "submit host" on which to initiate Grid jobs. User authentication credentials are required in the form of a Grid certificate that is authorized to use one or more Grid sites that provide computing and storage resources. The final component is access to a replica location service to catalog input and output files for a workflow.

The application programs to be executed are first described with virtual data language transformation statements, and these definitions are stored in a virtual data catalog. Once the transformations are cataloged, derivation (DV) statements to invoke them can be coded and cataloged in the VDC. Derivations are typically coded manually for initial tests, and then later generated on a larger scale using a script to process an entire data set. Languages such as Perl and Python are well suited for this task. Although generator scripts form an essential part of the current process, their use will gradually diminish as higher-level descriptive constructs are added to the VDL to iterate over various forms of application-specific data set structures.

Workflow "procedures" can be defined in virtual data language by grouping transformations together into workflow patterns that can be reused multiple times within a larger workflow. For example, to perform a multistage spatial filtering procedure on each file of a large data set, the filter pattern could be defined as a "compound transformation," which is then executed using one (opposed to many) derivation statement for each data set member.

From the definitions in a virtual data catalog, "abstract" workflows are then derived using graph traversal tools that use cataloged derivation definitions to create a graph of logical transformation invocations, creating a requested set of data files. This graph traversal recursively searches the catalog for definitions of the steps needed to produce the requested output data products, thus creating a specification of all the derivation steps needed to create the requested data. We call the output of this stage an "abstract" workflow because it is completely location independent: its steps are tied neither to any specific computing site nor to any specific physical file names

at these sites (either for data files or executable applications). A "concrete planner" is then applied to the abstract workflow and generates an executable workflow for either a local environment (typically for testing VDL) or for a Grid environment. The Virtual Data System planners that create Grid-executable workflows employ a variety of strategies for when and how they decide where in the Grid to run each derivation.

Certain workflows may consist of a huge number of steps, and may take many days, or, in some cases that we have worked on, many months, of Grid-wide computing to complete. Such large virtual data language workflows are often used in particle physics (Mambelli et al., 2004), astrophysics (Annis et al., 2002; Deelman et al., 2003), and genomics (Rodriguez et al., 2004). In recent work, including the explorations using fMRI described below, we employed prototype tools that provide higher-level interfaces for executing such workflows on the Grid. With these tools, once an abstract workflow is created, with the location-independent steps of a scientific process, then all remaining steps are automated: initiating execution on the Grid, tracking the progress of the workflow, and recording for diagnostic assessment any errors that occur.

Applications

To uncover and assess the issues involved in applying Grid virtual data techniques to typical computing problems in fMRI data analysis, we have conducted some preliminary experiments on Grid3 using the Virtual Data System. We focused our examinations on using raw neuroimaging data from the fMRI Data Center, on understanding how Grid resources could be leveraged to perform large-scale analyses that would be too time consuming on the resources available to most researchers in the field, and on determining how the procedures and derived data products from such efforts could be shared as a community resource. Three workflow patterns were chosen for more in-depth assessment: (1) the building of an averaged "template" brain, (2) a spatial normalization, and (3) an independent components analysis.

In our applications, we used application programs from the "Automated Image Registration" (AIR) package (Woods et al., 1998a, 1998b) for performing spatial realignment and warping of brain image data. A virtual data language transformation such as "AIR::alignlinear" was used to describe the various arguments used in a run of typical linear alignment. The input and output file arguments were specified in such statements, and where useful, default argument values were assigned. VDL transformations were created and cataloged for six different utilities from the AIR toolkit:

"align_warp," "definecommon_air," "alignlinear," "reslice," "reslice_warp," and "softmean," as well as several tools from FSL and other packages. The VDL "name space" capability (the string "AIR::" in the example above) was used to separate simple, unstructured tool names into different packages, both to avoid collisions and to serve as additional documentation and a convenient search key. VDL derivation statements were then created to execute the "alignlinear" tool on a specific set of input files. When executed, the output files from such derivations were listed in a replica catalog and an "invocation record" describing the run-time environment and resource utilization of the invocation was saved in the virtual data catalog.

Our sample data sets were drawn from files from two published and fMRIDC-catalogued studies: Buckner et al., 2000, and Head et al., 2004 (fMRIDC accession nos. 2-2000-1118W and 2-2004-1168X, respectively). Since our purpose was to explore analysis methodology using the Grid, rather than to test specific neuroscience hypotheses, we focused at this point in our assessment of virtual data solely on software tool usage modalities and on exploring how the fMRI research community might interact with the Grid tools. In some instances, synthetic data sets that mimicked the directory structure of the file system as well as the file types and sizes of the actual archived accessions under study were utilized for testing purposes.

Two data management tasks were required when moving from a locally executed virtual data language workflow to running on the Grid, and these were both assisted by utilities that we either enhanced or developed as part of our work. First, the application program (or suite of tools) needed to be installed at all Grid sites where we sought to execute. This was, as mentioned earlier, performed by a script that iterated over all (or set of) Grid sites. The second task involved making the input files of the workflow accessible to Grid jobs, which might be running at any site in the Grid, in a location-independent manner. This was accomplished by scripts that copied the input file set to a file system served by the Grid file transfer protocol, and then cataloged the files in the replica location server that served our study. This effort, too, was heavily automated and assisted by utility scripts that will be incorporated into the Virtual Data System, to make the copy-and-catalog steps as simple as an ordinary secure copy (scp) command from the SSH Communications Security tool set (Ylonen et al., 2001).

Our first sample application was a workflow for functional MRI analysis of the spatial normalization of data to a known brain template space. The input files for this workflow are high-resolution anatomical volumes. We created a workflow that would reorient and spatially

normalize the data to the standardized brain atlas from the Montreal Neurological Institute (MNI). Since the volumes needed to be in a similar orientation we selected a single study with a large number of anatomical volumes. This study (fMRIDC accession no. 2-2004-1168X) had four anatomical volumes per subject with 120 subjects. An example of a resulting spatially normalized image volume is shown in figure 7.4a. Using these same anatomical images in a second sample application, we designed a workflow to create an average normalized brain from the whole population of subjects. Figure 7.4b shows a less-detailed view of the directed acyclic graph for this process as well as the normalized "average brain" from this study.

Finally, in a third sample application, designed to demonstrate the processing throughput advantages of using the Grid, we constructed an independent components analysis workflow, which we then applied to a large fMRIDC data set (fMRIDC accession no. 2-2000-1118W) to extract interesting spatiotemporal features from the raw data (figure 7.5). Independent components analysis has proven to be a useful tool for understanding underlying functional systems in the brain, in particular, for assessing the role of baseline levels of brain activity (Greicius et al., 2004).

In our case, the use of ICA, though not necessarily intended to examine the brain systems underlying cognitively induced signal change in systems of distributed brain regions, was deemed to be a useful means for benchmarking Grid performance. We used MELODIC 2.0 (Beckmann and Smith, 2004), from the Oxford Center for Functional Magnetic Resonance Imaging of the Brain (FMRIB), as distributed as part of the FMRIB Software Library (FSL) tool suite. The MELODIC code requires a four-dimensional ANALYZE or NIFTI data set as input. We ran this workflow extracting 36 components from a 160-volume time course for a single subject in a study drawn from the fMRIDC archive: Buckner et al., 2000 (fMRIDC accession no. 2-2000-1118W-01).

To characterize the computational performance of this workflow, we compared its performance to that expected from a standard Unix (bash) shell script (table 7.2). Although the workflow and shell script have similar time scales for processing a single time course, the number of jobs performed in serial using a shell script would take a theoretical 162 hours (nearly a week) of processing time to produce a complete set of independent components analysis results for 30 such time courses. In contrast, using the Virtual Data System, regardless of the number of processing jobs submitted to the Grid, would take less than 6 hours. This is an obvious advantage of parallel versus serial computing, although it should be remembered

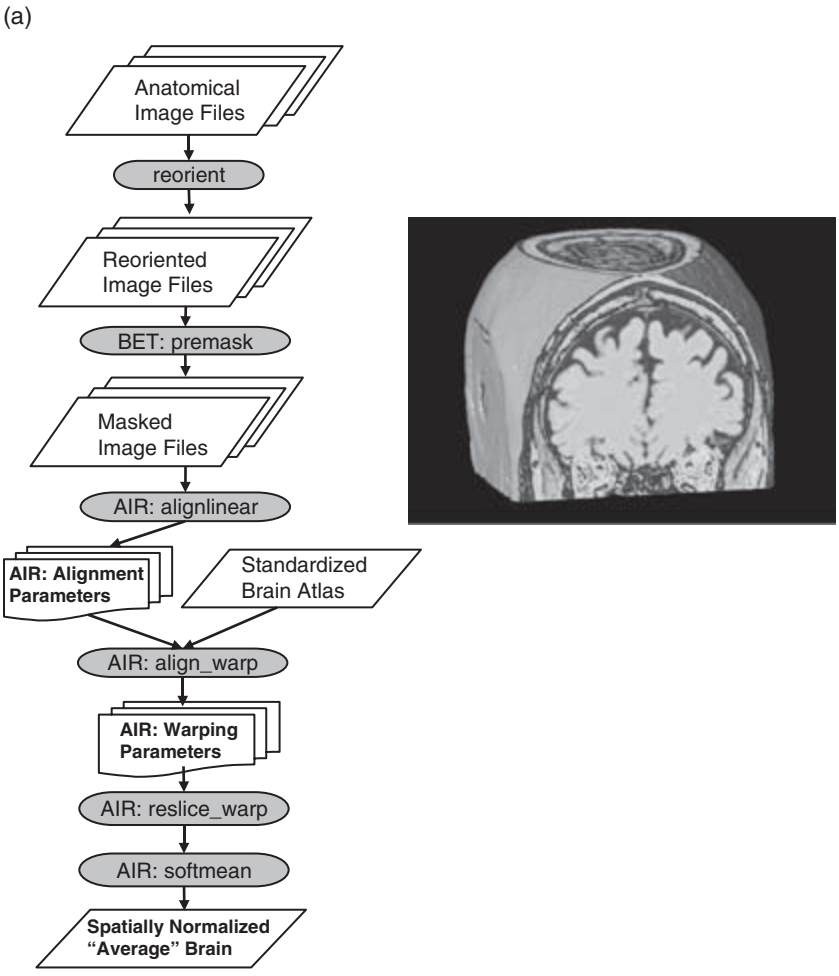
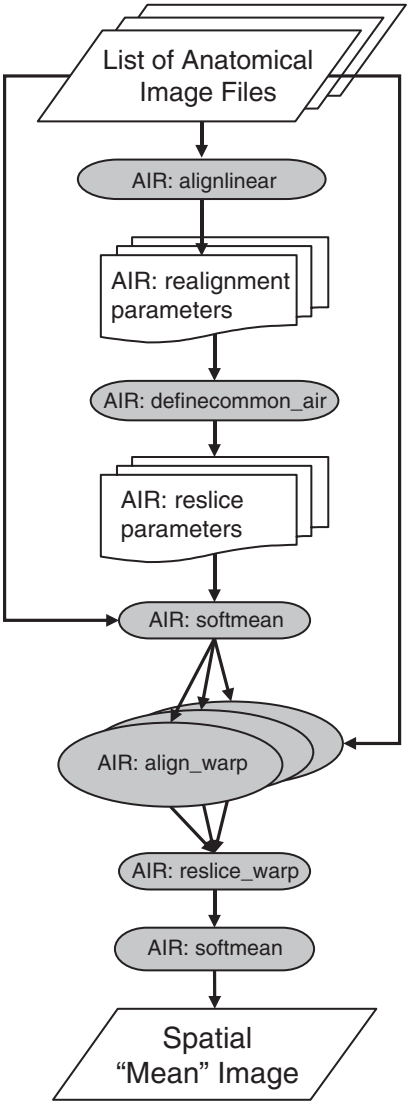


Figure 7.4
(*a*) Detailed directed acyclic graph (DAG) to spatially normalize a subject's anatomical brain image data to the Montreal Neurological Institute (MNI) brain template atlas. The cutaway anatomical template image was constructed using this workflow and data available from the fMRIDC. (*b*) A Higher-level view of a workflow DAG to spatially normalize multiple brain image volumes to the MNI template atlas and create a population-based average. Only the first few of the number of subject image volumes input to the workflow are shown here for illustrative purposes. The resulting "average brain" (with skull partially removed) is also shown.

(b)



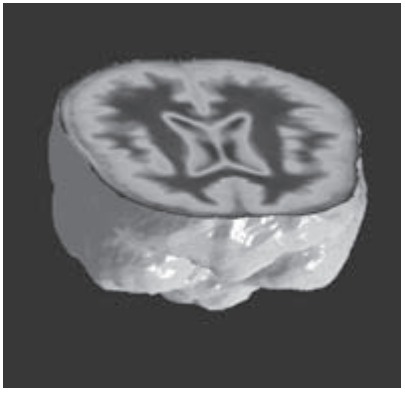


Figure 7.4
(continued)

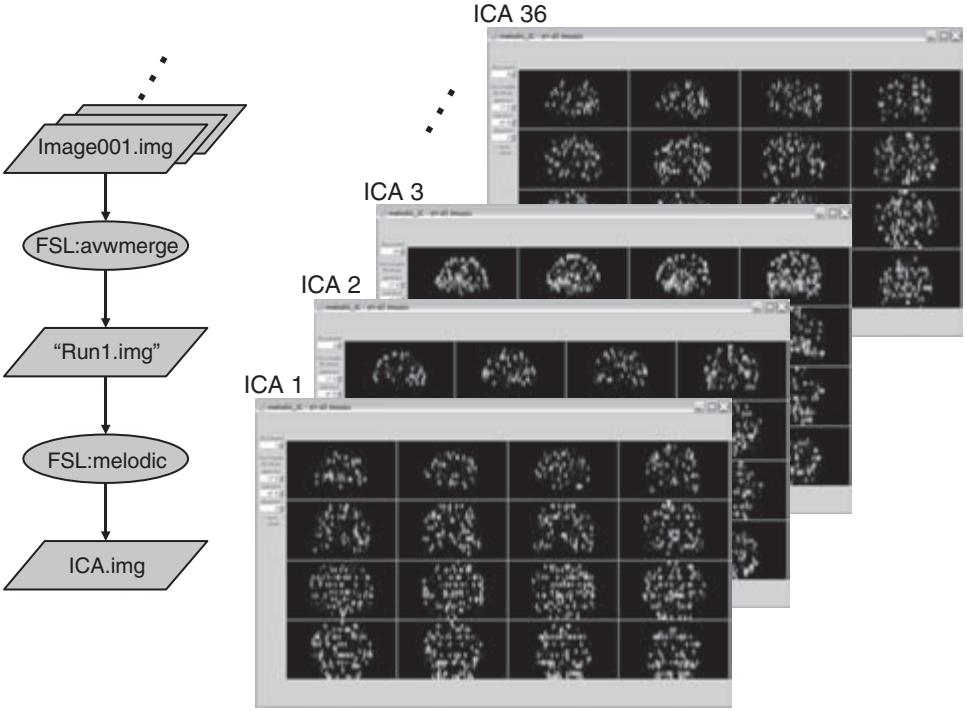


Figure 7.5
Directed acyclic graph (DAG) for performing an independent components analysis (ICA) using the FMRIB Software Library (Oxford) Melodic 2.0 routine and for use in benchmarking Grid performance on neuroimaging data. Any Analyze- or Nifti-compatible image viewer, such as the Java-based, Web-enabled viewer from the fMRIDC, can be used to view output images.

Table 7.2
Speedup chart for FMRIB Software Library independent components analysis using the Grid (hours : minutes)

Number of Processing Jobs	Shell Script Duration	Estimated Virtual Data System Duration
1	5:24	5:28
10	54:00	5:28
20	108:00	5:28
30	162:00	5:28

Timings based on 2.8 GHz Intel Xeon, 512 MByte RAM processors. Dataset was a single fMRI run having 160 volumes, $79 \times 95 \times 68$ voxels, 16-bit integer precision. It was assumed that (1) all systems were identically configured; (2) all queued jobs were executed immediately; and (3) the time needed to transfer data was identical between Grid sites. Virtual Data System overhead: Condor-G + Globus GRAM + SGE overhead: 90 seconds; pre- + postprocessing (for the selected data set): 160 seconds.

that the computational hardware for running the VDS does not reside locally and is available only as a consequence of using Grid. The Virtual Data System also has the benefits of queries, provenance tracking, parameter specification, as well as having the ability to create "compound workflow functions," whereas traditional shell scripting may not.

Limitations

The following observations from our experience with using virtual data techniques for fMRI analysis address both the potential advantages of the approach and the next steps we have identified for making the approach useful to the fMRI research community.

Scalability and Location-Independent Computing

Perhaps our most important finding is that the data and parameter encapsulation provided by virtual data language and the virtual data approach lends itself quite well to a powerful and valuable model of "location-independent computing." Specifically, when a sequence of analysis steps in a science workflow is described in VDL rather than in an ad hoc script, the precise declarations of the workflow's input and output files enable the use of automated "planner" and "executor" services. These Virtual Data System components handle all the tedious, error-prone work involved in dynamically selecting an execution locale for each step of the workflow. They automatically perform all of the explicit load balancing, data moving, data lookup, and cataloging actions required in a distributed environment.

Data Provenance Queries

Once studies have been run on the Grid using the Virtual Data System, a great deal of useful information on workflow structure, tool relationships, and data derivation attributes can be found in the virtual data catalog. From answers to specific queries, researchers can determine what parameters were used in deriving the datum in question, discover the derivations employed, and use this information to assess the derived object's validity. For example, they can ascertain whether a specific result was derived from an input brain image that had been anonymized, or had been registered by a linear alignment using a specific affine model.

Processing Data Sets

Presently, the virtual data language operates on individual files. To operate on structured, multifile or multidirectory data sets such as those in the fMRIDC archive, which are common throughout the field of fMRI and imaging studies in general, we frequently employ a "generator"—a script typically written in a higher-level scripting language such as Perl or Python. A generator serves, in part, as a "higher-level transformation": given a data set (a subset of the files in a directory tree), it performs a Virtual Data Toolkit transformation on each member of the data set, and operates by producing VDL *derivations* with the appropriate argument substitution.

File- and Directory-Naming Conventions

An area critical to the fMRI data set model is to establish natural, uniform conventions for the expression and manipulation of the file names and directory paths with a large archive and among the intermediate and final outputs of the workflow. To facilitate the use of fMRIDC data sets on the Grid, a flexible but general set of file name conventions is needed, one that can serve the entire user community (e.g., Nifti).

Community Setup and Usability

From our efforts, a picture is emerging of the type of community infrastructure needed to bring the benefits of virtual data methods and Grid computing to fMRI researchers. A logical next step in our continuing studies will be to Grid-enable the fMRIDC archival file server with the Grid file transfer protocol (GridFTP) service. An initial installation of this has been completed, which will enable us to grant secure and high-speed access to the entire fMRIDC archive with the GridFTP. A dedicated fMRIDC replica location service is being established that will enable portions of the archive to be cached throughout a Grid such as Grid3. As the studies proceed in parallel, significant network data transfer reduction can thus be expected from this caching model.

Our explorations have demonstrated great benefits from the automated site selection and load balancing mechanisms that are part of various research tools within the Virtual Data System. Preliminary work on predicting completion times for workflows will be highly valuable to science users. Continued enhancement of such fault-tolerant recovery mechanisms

is critical in a Grid environment, as are diagnostics that permit end users and administrators to easily determine what may have failed in a long-running workflow, and what corrective actions are required.

Integration

With the collection of neurobiological data into large databases and the availability of Grid-enabled means for large-scale data processing, a new form of discovery-oriented neuroscience is on the horizon. Researchers are increasingly wanting to examine vast and disparate collections of data in their hunt for unseen patterns that might provide clues to underlying biological mechanisms. By providing input for pattern-seeking and other relevant algorithms (see Ma, Tromp, and Li, 2002; Jones and Swindells, 2002; Schutte et al., 2002), the terabytes of data being collected in many fields can provide further insights into complex processes. The well-known successes of molecular biology, biomedicine, and astrophysics infrastructures have enabled experts in computer science, mathematics, and statistics to make significant contributions to these fields, from which most of their expertise would otherwise have been excluded. The need to build on these successes and to extend them to a wider set of scientific research arenas is an ever-present theme (see Altman, 2003; Brookes, 2001; Persson, 2000).

A discovery science for brain function will undoubtedly be located at the interface between large-scale archives of primary data and Grid computing capability. Indeed, we contend that functional neuroimaging could become a leading example of this phenomenon, whereby patterns of brain activity present across multiple subjects and dozens of studies can be systematically extracted and examined using Grid-based tools. The data from a published fMRI experiment contributed to a public repository such as the fMRIDC can be directly examined using Grid-capable tools by students and researchers who devise new and more sophisticated algorithms for analyzing the functional time courses. But where to begin? How can we move toward this vision?

In the first place, it is not difficult to envision the development of collections of Virtual Data System–based MRI processing tools that enable users to easily analyze data from the fMRIDC study data archive. The VDS would provide the virtual data catalog for many commonly used neuroimaging data transformation and derivation definitions, as well as basic scripts to instantiate many commonly used multistep data-processing pipelines (e.g., three-dimensional image registration). Utilities would be

provided to users so that the might easily set up Grid workflow runs, transfer and catalog fMRI datasets, and monitor remote program execution. Users could draw from the library of virtual data scripts provided or contribute their own virtual data transformations and scripts with accompanying directed acyclic graphs. More than just a repository for code, virtual data language pipelines performing like operations (e.g., spatial normalization) could be benchmarked for speed and accuracy and ranked accordingly. Those most highly ranked could become the "industry standards," determined not by committee, but by community, through usage statistics and rigorous assessment across multiple published data sets. In contributing their published studies to the fMRIDC, rather than providing a full-fledged data-processing narrative, authors could simply provide the virtual data language transformations, derivations, and scripts to the archive. Thus each study would have not only gigabytes of data, but also a ready means of replicating the exact analysis steps used by the original study authors. Variations on the order of transformations or pipelined operations could be tried to assess the properties of the resultant statistical outcomes (see, for example, LaConte et al., 2003). Finally, virtual data language scripts could also maximally leverage study metadata to process data contingent on various study, subject, or protocol parameters. For instance, the spatial normalization VDL script could be weighted by subject age and the data processed using an age-stratified brain template atlas. In this way, subject structural image volumes would be "normalized" to age-appropriate brain templates rather than some, supposedly, "ageless" stereotactic atlas.

As for computing resources, users already have what it takes to begin such Grid-level analyses now, with no need for large computers and massive disk arrays per se. Virtual data language–based algorithms are run on "virtual" fMRI data and submitted to the Grid, which transparently distributes the data-processing workload to geospatially distributed computers.

Rapid processing of large quantities of data in this way will lead to new scientific outcomes and patterns of results not envisioned through the examination of each study individually. Such patterns can suggest fundamental mechanisms, and the mechanisms can, in turn, suggest testable biological experiments to foster new hypothesis-driven research. Confirmed mechanisms add to the knowledge base of neurobiological science and provide the basis for further experimentation and the generation of still more valuable data that can be included in still greater analyses. This greater knowledge about fundamental cognitive processes will then suggest new and testable hypotheses that will lead to novel fMRI experimentation, the

data from which will be contributed back into the publicly available archive.

Such a cycle of science (figure 7.6) thus becomes a driving force behind the research endeavor and accelerates the growth of accumulated knowledge. It fosters greater use of costly and complex sets of brain imaging data, broadens the scope of the findings published in the literature, and serves to educate the next generation of neuroscientists.

Conclusion

Publicly available archives of primary fMRI data and distributed Grid computing are already playing important roles. The interface between these two domains promises unique neuroscientific outcomes giving rise to discovery science and broadening the scope of education. Moreover, with further refinements to standards and protocols for Grid computing, many forms of fMRI analyses may soon not be run on local machines at all, but will instead be performed using the resources of many geographically distributed computers. With more data present in these valuable archives and sophisticated Grid-based applications, greater opportunities for the genesis of new hypotheses and real scientific progress are just around the corner.

Acknowledgments

The fMRI Data Center (fMRIDC) Project is supported by a P20 grant from the National Institute of Mental Health Human Brain Project. The Grid Physics Network (GriPhyN) and the International Virtual Data Grid Laboratory (iVDGL) are supported by the National Science Foundation; the Particle Physics Data Grid (PPDG) and the Globus Alliance are supported by the U.S. Department of Energy Office of Science. We wish to thank the following persons and institutions for their contributions: fMRIDC team members Sarene Schumacher, John Wolfe, Autumn Agnoli, Michael Schmitt, Bennet Vance, Katherine Clemans, and Wendy Starr; Luiz Meyer, Federal University of Rio de Janeiro (UFRJ); the University of Chicago; GriPhyN and iVDGL team members Ewa Deelman, Gaurang Mehta, and Karan Vahi at the Information Services Institute, Marina del Rey, California; Miron Livny and Alan Roy at the University of Wisconsin; GriPhyN, iVDGL, and PPDG management team members Paul Avery, Rick Cavanaugh, Ruth Pordes, Doug Olson, and Richard Mount; and iVGDL operations team members Leigh Grundhoefer and Jorge Rodriguez. We also

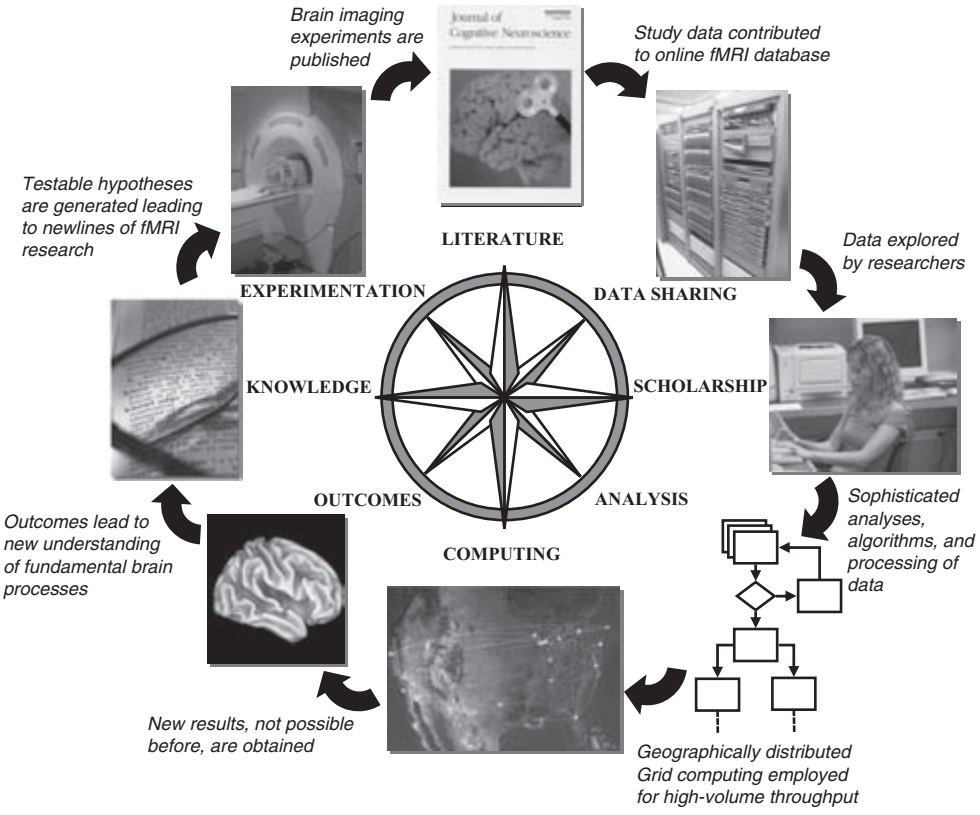


Figure 7.6
Cycle of a discovery science for neuroimaging made possible using large-scale data archiving and Grid-based computing. The availability of complete study data in online archives is enhanced by researchers being able to rapidly create novel processing methods for new analyses that utilize the Grid. Outcomes from these analyses broaden the collective knowledge base of fundamental brain function, may lead to new publications in their own right, and foster novel avenues for research and fMRI experiments. Data from these analyses are contributed to the archive and the cycle begins a new.

wish to thank Scott Grafton of the Dartmouth Brain-Imaging Center for consultation on the design of spatial normalization workflows; Natalia Maltsev, Alex Rodriquez, Dinanath Sulhake, and Veronica Nefedova for applying the GriPhyN Virtual Data System to genomics research; Jim Annis, Neha Sharma, Vijay Sekhri, and Michael Milligan for their Virtual Data System work on the Sloan Digital Sky Survey; Albert Lazzarini, Scott Koranda, Kent Blackburn, and many others for their work on the Laser Interferometer Gravitational Wave Observatory (LIGO) project; Ed May, Jerry Gieraltowski, Marco Mamelli, Rob Gardner, and Yuri Smirnov for their work on ATLAS (A Toroidal Large hadron collider Apparatus).

References

Altman, R. B. (2003). The expanding scope of bioinformatics: Sequence analysis and beyond. *Heredity*, *90*, 345.

Annis, J., Zhao, Y., Voeckler, J., Wilde, M., Kent, S., and Foster, I. (2002). Applying Chimera virtual data concepts to cluster finding in the Sloan Sky Survey. In *Proceedings of Supercomputing 2002*. Baltimore.

Beckmann, C. F., and Smith, S. M. (2004). Probabilistic independent component analysis for functional magnetic resonance imaging. *IEEE Transactions on Medical Imaging*, *23*, 137–152.

Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., and Wheeler, D. L. (2003). GenBank. *Nucleic Acids Research*, *31*, 23–37.

Berman, H. M., Battistuz, T., Bhat, T. N., Bluhm, W. F., Bourne, P. E., Burkhardt, K., Feng, Z., Gilliland, G. L., Iype, L., Jain, S., Fagan, P., Marvin, J., Padilla, D., Ravichandran, V., Schneider, B., Thanki, N., Weissig, H., Westbrook, J. D., and Zardecki, C. (2002). The Protein Data Bank. *Acta Crystallographica*, *D58*, 899–907.

Brookes, A. J. (2001). Rethinking genetic strategies to study complex diseases. *Trends in Molecular Medicine*, *7*, 512–516.

Buckner, R. L., Snyder, A. Z., Sanders, A. L., Raichle, M. E., and Morris, J. C. (2000). Functional imaging of young, nondemented and demented older adults. *Journal of Cognitive Neuroscience*, *12* (suppl. 2), 24–34.

Catlett, C., and Smarr, L. (1992). Metacomputing. *Communications of the ACM (Association of Computing Machinery)*, *35*, 44–52.

Deelman, E., Kesselman, C., Mehta, G., Meshkat, L., Pearlman, L., Blackburn, K., Ehrens, P., Lazzarini, A., Williams, R., and Koranda, S. (2002). GriPhyN and LIGO, Building a Virtual Data Grid for Gravitational Wave Scientists. *Proceedings of the 11th IEEE International Symposium on High Performance Distributed Computing (HPDC'02)*, p. 225, http://csdl2.computer.org/persagen/DLAbsToc.jsp?resourcePath=/dl/proceedings/&toc=comp/proceedings/hpdc/2002/1686/00/1686toc.xml&DOI=10.1109/HPDC.2002.1029922

Foster, I. (2003). The Grid: Computing without Bounds. *Scientific American*, *288*(3), 78–85.

Foster, I., and Kesselman, C. (1998). Globus: A metcomputing infrastructure toolkit. *International Journal of Supercomputer Applications*, *11*(2), 115–129.

Foster, I., Kesselman, C., and Tuecke, S. (2001). The anatomy of the Grid: Enabling scalable virtual organizations. *International Journal of Supercomputer Applications*, *15*(3), 200–222.

Foster, I., Voeckler, J., Wilde, M., and Zhao, Y. (2002). Chimera: A virtual data system for representing, querying, and automating data derivation. In *Fourteenth International Conference on Scientific and Statistical Database Management*. Edinburgh.

Frey, J., Tannenbaum, T., Foster, I., Livny, M., and Tuecke, S. (2002). Condor-G: A computational management agent for multi-institutional Grids. *Cluster Computing*, *5*, 237–246.

Greicius, M. D., Srivastava, G., Reiss, A. L., and Menon, V. (2004). Default-mode network activity distinguishes Alzheimer's disease from healthy aging: Evidence from functional MRI. *Proceedings of the National Academy of Sciences*, *USA*, *101*, 4637–4642.

Head, D., Snyder, A. Z., Girton, L. E., Morris, J. C., and Buckner, R. L. (forthcoming). Frontal-hippocampal double dissociation between normal aging and Alzheimer's disease. *Cerebral Cortex*.

Jones, D. T., and Swindells, M. B. (2002). Getting the most from PSI-BLAST. *Trends in Biochemical Science*, *27*, 161–164.

LaConte, S., Anderson, J., Muley, S., Ashe, J., Frutiger, S., Rehm, K., Hansen, L. K., Yacoub, E., Hu Xiaoping, Rottenberg, D., and Strother, S. C. (2003). The evaluation of preprocessing choices in single-subject BOLD fMRI usings NPAIRS performance metrics. *Neuroimage*, *18*, 10–27.

Ma, B., Tromp, J., and Li, M. (2002). Pattern Hunter: Faster and more sensitive homology search. *Bioinformatics*, *18*, 440–445.

Mambelli, M., Gardner, R., Smirnov, Y., Zhao, X., Gieraltowski, G., May, E., Vaniachine, A., Baker, R., Deng, W., Nevski, P., Severini, H., De, K., McGuigan, P., Ozturk, N., and Sosebee, M. (2004). ATLAS data challenge production on Grid 3. Paper delivered at the Computing in High-Energy Physics conference, 2004 (CHEP'04). Interlaken, Switzerland.

Persson, B. (2000). Bioinformatics in protein analysis. *Exs*, *88*, 215–231.

Rodriguez, A., Sulakhe, D., Marland, E., Nefedova, V., Wilde, M., and Maltsev, N. (2004). Grid-enabled server for high-throughput analysis of genomes. Paper delivered at the Workshop on Case Studies on Grid Applications. Berlin.

Schutte, B. C., Mitros, J. P., Bartlett, J. A., Walters, J. D., Jia, H. P., Welsh, M. J., Casavant, T. L., and McCray, P. B., Jr. (2002). Discovery of five conserved beta-defensive gene clusters using a computational search strategy. *Proceedings of the National Academy of Sciences*, *USA*, *99*, 2129–2133.

Van Horn, J. D., Grafton, S. T., Rockmore, D., and Gazzaniga, M. S. (2004). Sharing neuroimaging studies of human cognition. *Nature Neuroscience*, *7*, 473–481.

Van Horn, J. D., Grethe, J. S., Kostelec, P., Woodward, J. B., Salam, J. A., Rus, D., Rockmore, D., and Gazzaniga, M. S. (2001). The fMRIDC: The challenges and rewards of large-scale data basing of neuroimaging studies. *Philosophical Transactions of the Royal Society*, *London*, *B356*, 1323–1339.

Woods, R. P., Grafton, S. T., Holmes, C. J., Cherry, S. R., and Mazziotta, J. C. (1998a). Automated image registration. 1. General methods and intrasubject, intramodality validation. *Journal of Computer-Assisted Tomography*, *22*, 139–152.

Woods, R. P., Grafton, S. T., Watson, J. D., Sicotte, N. L., and Mazziotta, J. C. (1998b). Automated image registration. 2. Intersubject validation of linear and nonlinear models. *Journal of Computer-Assisted Tomography*, *22*, 153–165.

Ylonen, T., Kivinen, T., Saarinen, M., Rinne, T., and Lehtinen, S. (2001). SSH transport layer protocol. Internet Engineering Task Force. Internet draft.

Ylonen, T., Kivinen, T., Saarinen, M., Rinne, T., and Lehtinen, S. (2001). SSH Protocol Architecture. *Proceedings of the Internet Engineering Task Force. Report from the Network Working Group*. Available online at: http://www.letf.org/proceedings/Olmar/index.html.