(1)

# Volume I, Technical and Management Proposal

(2) *BAA number*: DARPA-BAA-14-59
(3) *Technical area*: TA1+TA2

(4) *Proposal Title:*

## From RAGs to Riches: Utilizing Richly Attributed Graphs to Reason from Heterogeneous Data

(5) *Proposer's Reference Number*: N/A

(6) *Lead Organization*: Johns Hopkins University

(7) *Type of organization*: Other Educational

(8) *Technical Point of Contact*: Dr. Joshua T. Vogelstein
3400 N. Charles St., 301 Clark Hall
Baltimore, MD 21218
Phone: (443) 858-9911
Email: jovo@jhu.edu

(9) *Administrative Point of Contact*: Ms. Alison Wampler
400-W Wyman Park Building
Baltimore, MD 21218-2686
Phone: (410) 516-7306; Fax: (410) 516-7775
Email: awampler@jhu.edu

(10) *Total funds requested*: $1,975,863

(11) *Award Instrument Requested*: Grant

(12) *Place and Period of Performance*: Johns Hopkins University
3400 N. Charles St.
Baltimore, MD 21218

Phase I: 3/01/15–5/31/16
Phase II: 6/1/16–5/31/17
Phase III: 6/1/17–5/31/18

(13) *Other Team Members*: None

(14) *Date Proposal Was Prepared*: November 6, 2014

(15) *Proposal Validity Period*: Mar 1, 2015 – May 30, 2018

## Contents

## List of Figures

## List of Tables

# I Executive Summary

### What are you trying to do? Articulate your objectives using absolutely no jargon.

Historically, the primary bottleneck in neuroscience was data collection, now it is data analysis. We will move the bottleneck to data modeling modeling. We will achieve this via building open source Web-services that automate reference pipelines for three different experimental modalities spanning spatiotemporal scales: whole-brain CLARITY, large-scale optophysiology, and multimodal MRI. These Web-services will enable 1-click processing converting raw data into neuroscience "objects" of semantic meaning, such as regions, cells, and processes, suitable for analysis.

### How is it done today, and what are the limits of current practice?

If the data are small enough, each laboratory develops its own pipelines, typically using neuroscience students for software engineering, and limiting analysis to their own data. Code is not properly documented nor open, and only runnable by the developer herself, such that even within a lab, there will be multiple different pipelines, without comparison capabilities. For big data, labs do not even have their own pipelines, because they lack both the computational resources and expertise to implement them. Thus, data lies idle, or is never even collected because the lab heads know that the lab will be unable to analyze and therefore, utilize the data.

### What is new in your approach and why do you think it will be successful?

We will deploy reference open source pipelines using the principles of "continuous integration" on serverless computing models (such as AWS Lambda), applied to reference data that we will make open access. Open source pipelines would obviate the need for labs to develop their own pipelines. Deploying them on serverless computing models eliminates the need for labs to purchase and maintain custom clusters for data analysis. Using continuous integration will enable the work to be fundamentally reproducible and extensible. And developing them on reference datasets that we make open access will provide example use cases and demonstrate success, in addition to bringing fully processed data to anybody. This will be successful because we will simultaneously eliminate all the primary bottlenecks to developing models on state-of-the-art neuroscience data, and we have preliminarily demonstrated the ability to develop prototype pipelines.

### Who cares? If you succeed, what difference will it make? What are the risks?

Anybody collecting CL, Ophys, or M3RI data will care immediately, others will have these success stories to build upon. The main risk is that neuroscientists are accustomed to a particular scientific workflow, and this will be a disruptive technology. That said, we have historical examples of related disruptive technologies (e.g., the Sloan Digital Sky Survey in cosmology), that caused a paradigm shift in the way science was conducted.

### How much will it cost? How long will it take?

Each modality will require a professional software engineering, a post-doctoral fellow, cloud computing resources, a local cluster, part-time support staff for users, travel to collaborators, and publication fees, totaling $250,000 (direct) per year per modality, for a total of three years.

### What are the mid-term and final exams to check for success?

We will develop "bare-bones" examples by the end of year one, fully operational reference pipelines by the end of year two, and much improved (via community feedback) pipelines by end of year three.

# II Executive Summary Slide

# III Goals and Impact

## III.A Overview

The BrainLab CI prototype system will deploy an experimental-management infrastructure that allows users to construct community-wide experiments that implement data and metadata controls on the inclusion and exclusion of data. It will do so by building upon the principle of continuous integration (CI), adopted from the agile software development community. The CI platform allows the designer of an experiment to place multiple simple or complex controls on data, such as requiring specific metadata, that data are registered to a given atlas, or that data are collected using specific experimentation protocols. Controls may include reprocessing data to make it compliant, e.g. reregistering MRI data to a given atlas or band-pass filtering electrophysiology data. Controls and processing will run user-defined scripts and programs within a software container (e.g., Docker) on a serverless computing cloud, such as Amazon Lambda. Containers encapsulate all dependencies to ensure that software runs uniformly against all inputs on all platforms. Serverless computing automatically and elastically provisions cloud resources to run containers.

BrainLab CI will create a meaningful way to scale science from the tens of subjects used by individual labs today to thousands or tens of thousands, while allowing scientists to exert controls that ensure data quality, initially with two different experimental patterns: (1) An incremental experiment defines an experiment against a existing data set and then opens the experiment to community contributions. Other labs/scientists contribute data which gets processed on submission and included or rejected. The experiment maintains online dashboards that show how additional data change results with complete provenance. (2) A derived experiment forks/branches an existing experiment allowing a researcher to change properties, such as an acceptance criteria or analysis algorithm, but otherwise run the same pipeline against the same inputs. Full provenance allows scientists to reason and debate about how modifications affect outcomes. These patterns can be composed and the fork/branch version control model of the CI system connects changes in experiments to outcomes. We will co-develop community experiments for MRI and for neurophysiology (including both optical and electrical physiology); these domains were chosen because they are critical to NSF BRAIN neuroscience, have large bodies of unshared data living in silos in different labs, and they produce large numbers of relatively small (GB-scale) data sets that are managed easily in a prototype.

**Broader Impacts** BrainLab CI has the potential to permanently transform practice in neuroscience. The design principles for the system grew out of the NSF KAVLI Global Brain Workshop's discussion on realizing a global-shared infrastructure for scientific discovery. BrainLab CI overcomes major obstacles to data sharing. It does not restrict scientists to a specific workflow system, ontologies, software frameworks, etc. Rather, scientists package their custom experiments into containers and set admission criteria for new data. Scientists share data without losing control over data quality; they gain full provenance on how all subsequent experiments use their data and algorithms. This provenance can be used to debate and reason about different results from the same data or how specific data changes affect results. With these important barriers to data sharing removed, we envision a system that meaningfully integrates the 1000s of publicly available data resources in MRI and neurophysiology and creates incentives for data sharing for individual labs and data collectors, who gain great analytic power by confronting their new data with a massive corpus.

**Intellectual Merit** BrainLab CI adopts cutting-edge practice in agile software development and cloud computing to build a totally unique capability for neuroscience. By adopting containers, which encapsulate all software dependencies, and serverless computing, we will create an automated system that runs arbitrary and sophisticated checks on submitted data. Data contributors experience no overhead: they upload a data set via file transfer or Web service and the system automatically runs the continuous integration suites. Experiment designers write simple programs, e.g. scripts, or complex workflows to express their constraints and continuous integration enforces these automatically and at scale.

## III.B Description

The brain research community desperately needs to build shared infrastructure that combines the many publicly available imaging data sets and allows researchers to define and conduct experiments that confront thousands or tens-of-thousands of subjects, rather than the tens used today by individual labs. Scientists will gain great analytical power by referencing their studies against a massive corpus, allowing them to focus on experimental design and data analysis, rather than data collection. This will engage a national and international community of researchers that do not collect—an expensive activity limited to few labs. It will also drive a change in scientific culture, encouraging data sharing, the development of common analysis tools, and accelerated discovery from connecting ideas, tools, data, and people.

Data quality and data control present major obstacles to successful data sharing and to participation. Bad data leads to bad science and data collectors and the neuroscience community are legitimately concerned about the misuse of data. To be useful in a shared infrastructure, datasets need to have sufficient metadata and employ collection, preparation, and data processing protocols that produce comparable data. Examples include stimulus in functional MRI studies or cell type and sample preparation for single-cell spike-train data. For a shared infrastructure to be transformative, it needs to link experiments to data, allow data providers to exert control on how data are used and what data can be included in a study, and allows data consumers to define and customize experiments consistent with exploratory analysis and data mining.

We propose a platform to define ongoing, incremental, and community experiments with data controls based on the agile software development principle of *continuous integration* (CI). In CI, every time new source is committed, it is automatically built against a suite of tests and multiple configurations and deployed with the goals of keeping all contributions merged, maintaining a single view of the repository, and keeping deployed software updated. We propose an analogous workflow for neuroscience experiments in which (1) the community contributes new data incrementally to an experiment and (2) scientists derive and customize new virtual experiments that inherit existing data and data controls.

The resulting *BrainLab CI* will take existing studies and provide a continuously updated view that integrates the latest results, couples changing results to new data, and provides dashboards that link outcomes to input data. Experiment maintainers gain provenance for outcomes and can refine data controls. An incremental community experiment will build on an established result, extending its scope and power. Researchers will seed the repository with reviewed and published results by pushing an experiment and a dataset to the BrainLab CI repository, for example, an Alzheimers study that links the shape of the Amygdala with functional MRI that produces a classifier that identifies diseased brains with a given accuracy and significance (p-value). The research will include scripts and programs that implement data controls that describe the data selection criteria: (1) requiring that shape data are registered to a given atlas, potentially running a registration pipeline for non-conforming data and accepting/rejecting data based on the result; (2) checking for metadata that describes a compatible imaging protocol, such as duration and stimulus; and, (3) validates that fMRI data were processed with a given pipeline (CPAC [1], FSL [2]), reprocessing the data into this form when possible. This experimental definition provides a base version and checkpoint that links the data set with the published result and repeatedly computes the published result in BrainLab CI.

Additional data sets are added to the experiment in batches or incrementally. Our example experiment may link to the thousands of compatible brains stored in MRICloud (http://mricloud.org). The outcome of linking will include updated experimental results based on the new data and a description of which data passed controls and why data were rejected. A continuous *community experiment* will accept new imaging raw data incrementally, registering and pre-processing it as part of data controls, and maintain an up-to-date view of outcomes as data changes. Each contribution defines a new experiment that has provenance, linking results to the source data. Analysis dashboards allow all to look at how results vary based on the inclusion/exclusion of different data. Controls may be added or removed to create experimental variants that reveal the structure of an outcome.

BrainLab CI inherits the fork/clone model of software repositories, allowing scientists to vary experiments and inherit from existing experiments. Experimental variants document the data controls so that researchers can reason about and dispute divergent findings and link the divergence to selection criteria, algorithms, etc. In our example, we may include diffusion MRI data and study the tradeoff in classification between adding a new imaging modality and reducing the number of subjects. A scientist exploring bias/variance tradeoffs would *fork* an existing experiment and change data selection criteria. The results would be directly comparable to the original and new data could be added to both experiments independently. The fork/clone/versioning model when combined with container-based computing (see Technology) guarantees repeatability and reproducibility. Programmatic data selection against online repositories create data provenance, specifying the exact inputs. The experimental version links data to an outcome. And, because we run in containers, the same program produces exactly the same output on future and legacy systems.

## III.C Use Cases

BrainLab CI targets three neuroscience problem domains: (1) multimodal (structural, functional, and diffusion) MRI imaging studies linked with shape, genomic, and phenotypic information, (2) neurophysiology data linked with cell type, genomic, and computational neuroscience models, and (3) whole-brain CLARITY data including immediate early gene expression and anatomical data. All three domains have a large body of unshared data from many labs that are currently incomparable. Each produces large numbers of relatively large (TB) data sets that can be combined in a modest computational infrastructure. Furthermore, both were identified as ideal candidates for sharing in the recent Global Brain Workshop (http://brainx.io).

### III.C.1 BrainLab MRI

An MRI experiment may consist of a diffusion and structural scan for a number of subjects using a novel set of scanning parameters. The data could be stored in a standard format accepted by many public repositories (http://bids.neuroimaging.io/) and ingested into the NeuroData system. Then, nonlinear image registration would warp the data to a common atlas of the user's choosing (currently, NeuroData stores over 24 atlases; http://docs.neurodata.io/nddocs/mrgraphs/atlases.html), followed by tensor estimation, tractography, and network estimation. The experimentalist may then seek to understand how the images generating under this new scanner sequence look as compared with previously existing sequences. Because many different datasets are stored in the same format, linking to others is easy. Figure 1 shows one of the many possible different summary statistics across eight different datasets, demonstrating significant batch effects for the KKI2009 data: its values are an order of magnitude higher than for the other datasets;

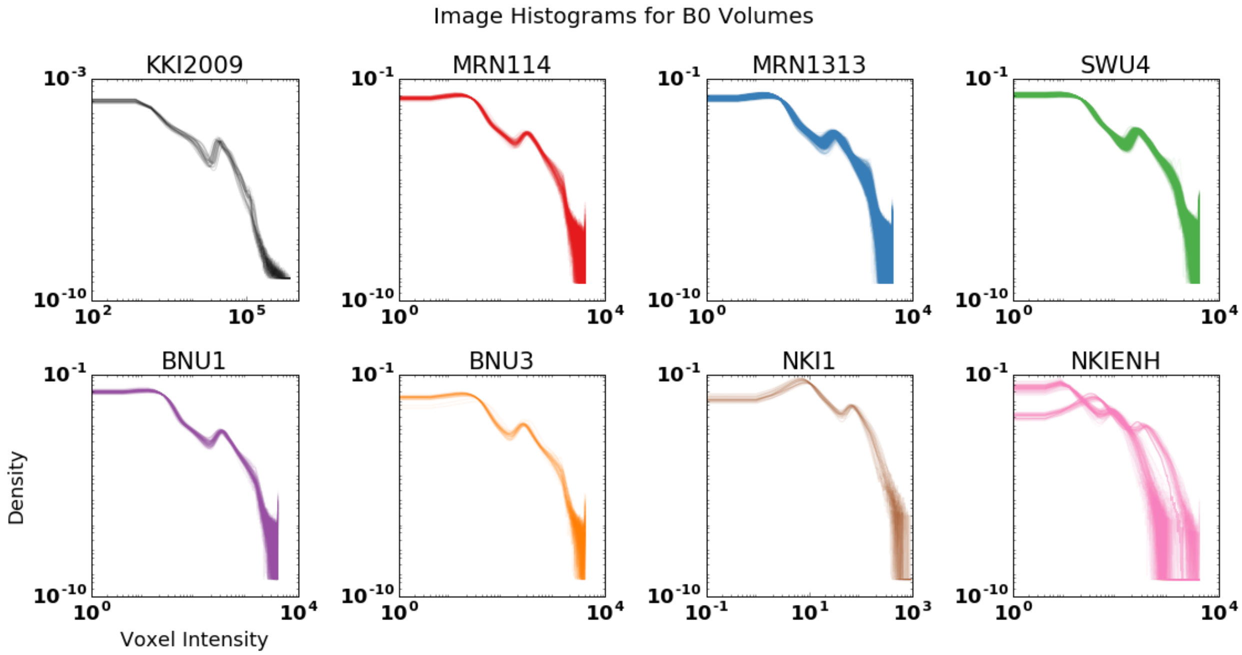


Figure 1: An example comparison of summary statistics across different MRI datasets.

checking the metadata reveals that KKI2009 used a different MRI machine than the other datasets in this comparison.

### III.C.2 BrainLab Physiology

A typical experiment from calcium imaging may proceed as follows. Image data are stored in a predetermined general specification (for example, images stored according to the NeuroData specification http://docs.neurodata.io/ndstore/sphinx/ingesting.html), ingested into the system, cell bodies are automatically detected, followed by spike detection, and then network estimation. In addition to the physiology data, there will also typically be metadata, which could be stored according to the Neurodata without Borders specification http://www.nwb.org/resources/. A single experiment might include multiple trials per "run", with multiple runs per animal, and multiple animals. For example, the trial specific metadata may be a time-varying visual stimuli for an anesthetized animal. In such a scenario, the experimentalist may desire to condition the network estimation on the stimuli, and look at the statistical network differences across conditions.

Under such a scenario, the dashboard may provide certain per run visualizations as depicted in Figure 2 (top three panels). Next, it may show summary statistics for said run (next panel). Finally, it could show the statistical comparisons pooling over all animals in the experiment (bottom panel).

In this experiment, of course, there are many different variables that different experimentalists may want to consider, or even the same experimentalist may want to consider at different times. For example, as new and improved spike detection tools become available [**?** ], users will want to swap out the old and swap in the new. Of note, if other people had also collected data from the same neurons (say, in an invertebrate species), with overlapping stimulus conditions, it would be easy to also pool that data. Finally, while this entire experiment has been discussed in the context of optophysiology based on calcium imaging, the number of changes required if one were to fork this to apply it to fMRI data are quite minimal, because of the common interface to data across modalities, laboratories, and experiments.

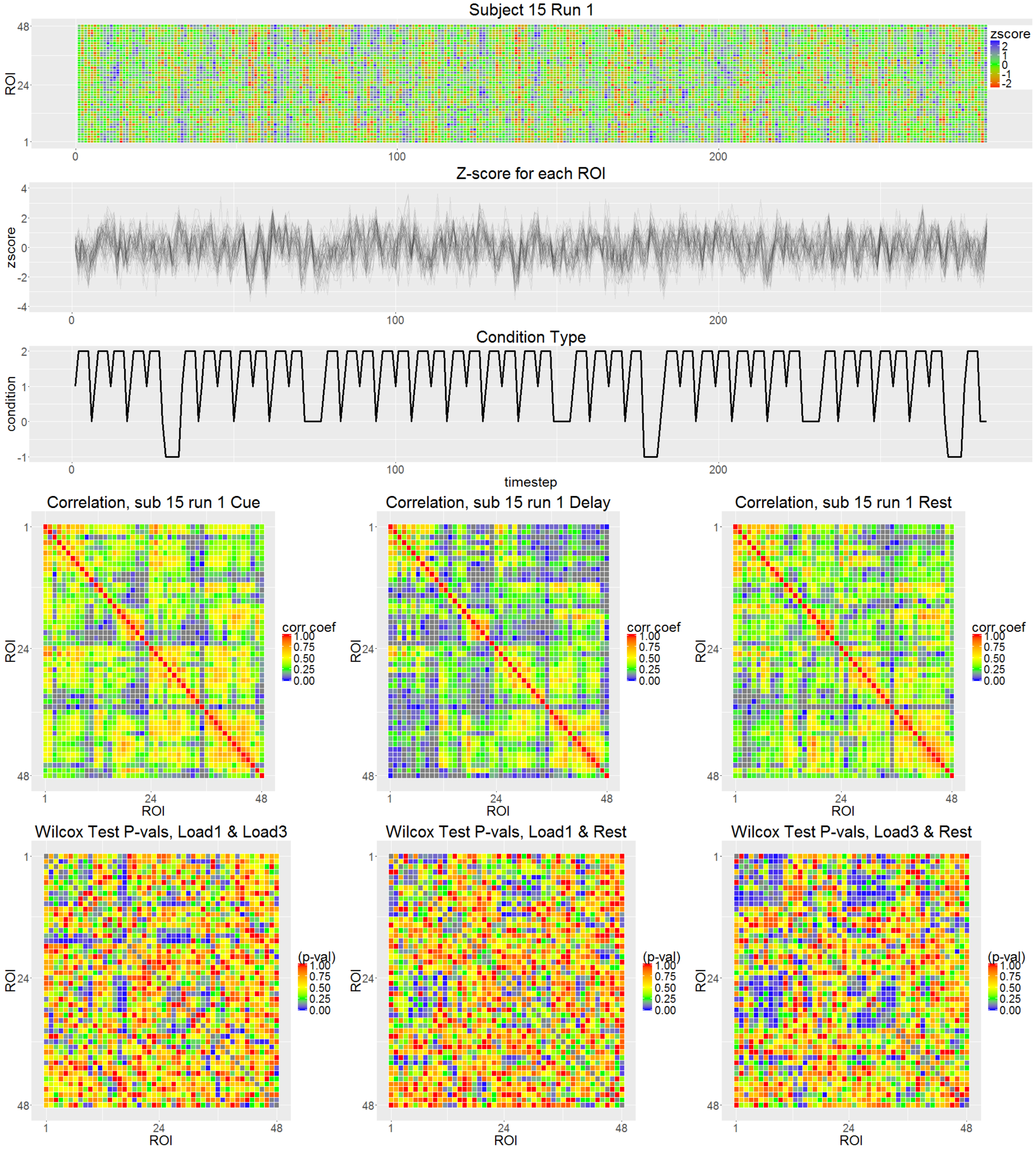
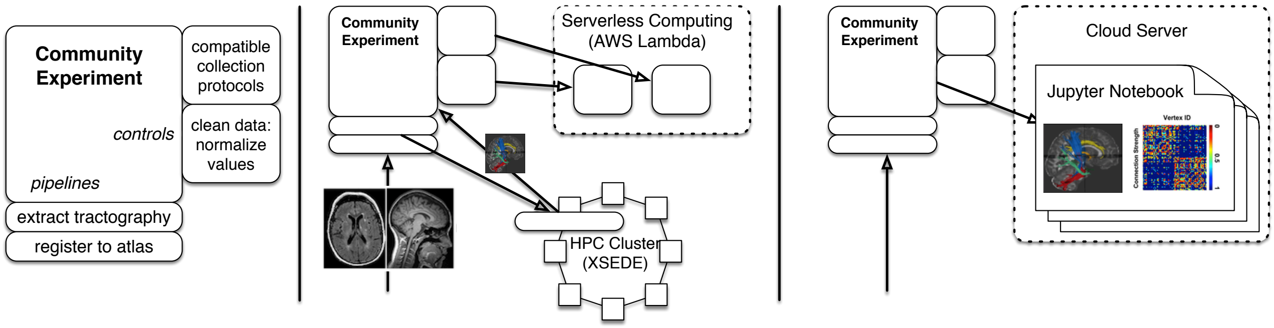


Figure 2: An example dashboard for an physiology experiment. Note that this dashboard could be effectively the same across a wide range of different experiments, spanning many different scales and domains.



(a) Definition of a community experiment with processing pipelines and data controls

(b) Uploading an MRI image extracts tractography, uploads derived data, and runs checks against MRI input data and pipeline output.

(c) An analysis session launches a server-side Jupyter notebook that provides visualization, statistics, and interactive analysis.

Figure 3: Overview of user interactions with a community experiment.

### III.C.3 BrainLab CLARITY

## III.D System Overview: Goal and Deliverables

The project will develop a continuous-integration platform on top of NeuroData [3] and MRICloud that provide version control, data provenance, integrated testing, automated experiments, and analysis dashboards. A fundamental capability will be running controls against new data sources and updating experiments based on new inputs. BrainLab CI will rely on software containers (e.g., Docker) to encapsulate the scripts and programs associated with a community experiment. Containers makes all scripts and programs *exactly repeatable* on different clusters and operating systems. We will integrate containers with event-based serverless computing, such as Amazon Lambda, so that uploading new data to a community experiments triggers data controls and processing. In this way, we can deploy BrainLab CI without any infrastructure investment, paying only for the computing that we use when we use it.

The project will **deliver** a prototype CI system that accepts both physiology and MRI data that:

- runs continuous experiments with community contributions and data acceptance criteria;
- supports inbound data processing based on serverless cloud computing as part of a data check in;
- supports refinement branch/fork of existing experiments;
- and, provides analyses and visualizations that reflect the outcome of the experiment for all versions.

We first describe the operation of the system, indicating how and when data processing and controls are triggered and run. We then connect software components to cloud and HPC architectures.

**Dataflow/Operational Architecture** The design of BrainLab CI inherits from the service-oriented architectures that build continuous integration and continuous deployment into software repositories, such as GitHub and bitbucket. Both maintain a *web hooks* API that invokes external services when events occurs; for BrainLab CI, the events of interest will be (1) when new data are submitted to an experiment, (2) when experiments are cloned or pulled, and (3) when experiments are forked or branched. We describe a few of the operations for a diffusion MRI example from the perspective of a neuroscientist running an experiment (Figure 3). We also connect each operation to the analagous git command.

Users may define a community experiment (git init) or derive a new experiments from an existing experiment (git branch or git fork) using the management console at BrainLab CI. (As with git, equivalent commands exist for automation and power users.) An experiment consists of a repository in which to place experimental data and metadata and a set of web hooks. For web hooks, users provide scripts or programs to be run on events that modify the experiment or its data. An experiment will also provide dashboards and vizualtions that generate and maintain the current state of results (graphs, tables, and figures). Experiment owners create Jupyter notebooks to produce these products that BrainLab CI runs automatically, as a web hook, when experimental data change.

When data are submitted to the users local version of a community experiment (git commit), BrainLab CI invokes registered web hooks. The example submits a NifTI file to the functional MRI experiment. The first web hook checks whether the file has sufficient metadata and a conforming data (resolution and data type). If the file passes input checks, the next Web hook executes a preprocessing pipeline by submitted a supercomuting job to a cluster, NSF XSEDE in this case. This computation converts MRI imaging data into tractography data by stripping the skull from the image, registering the data to the Desikan atlas [4], and extracting neural pathways through white matter. Tractography data are submitted back to the experiment and triggers a Webhook the updates local experimental result.

Subsequently, the user contributes these changes back to community experiment, a git push. The push Webhook updates the user's changes with the global view, in this case using the same update routine that runs locally. Each user keeps experimental results consistent with the version of the experiment on which they are working. Other user's contributions do not automatically update their working version. Users can update their experiment (git pull) to merge in community contributions. When they do so, a web hook will update their experiment.

Users may inspect and revert to past versions of the experiment (git checkout and git reset). For example, if the owner of an experiment notices that data updates have produced inconsistent results because data checks were insufficient. They may revert to a previous good version and add/modify the web hook routines that set criteria for data acceptance. Users that do not own the experiment and wish to change web hooks must fork or branch the repository.

Analysis dashboards and visualizations are continuously updated, but their presentation are not linked to Web hooks. Dashboards consist of graphs, statistics and visualizations that are defined in Jupyter notebooks—a complete R/Python/Julia environment implemented on the server. When users access a notebook, they run the analyses in the notebook, materializing the dashboard based on the most recent updates. For a community experiment, the dashboard runs on the server and presents a uniform view to all. For a forked, local or modified experiment the dashboard runs locally based on all committed updates, whether they have been pushed or not.

**System Architecture** We deploy BrainLab CI using a services-oriented architecture that eliminates systems management and minimizes costs (Figure 3). Users enter our system through a Web application that provides the git-like experimental management Web services and present analysis dashboards and visualizations. For our prototype, we plan to develop on Amazon Web Services (AWS), although the system design is simply portable to open-source clouds (OpenStack) and other providers (Google Compute Engine). The Web application scales elastically in an AWS scaling group, (de)commissioning servers based on demand.

We use serverless computing, with AWS Lambdas, for all continuous integration web hooks so that we pay only for the computing that we use. An AWS Lambda instantiates computing resources on demand in response to a Web service request, e.g. scripts that implement data controls on upload.

We link to supercomputers for compute-intensive tasks and long-running workflows, such as image processing, tractography for MRI, and cell detection and spike sorting for physiology. A Web hook will submit a supercomputing job through an HPC scheduler (qsub, msub, or srun) and we launch a management task with the AWS Elastic Container Service to monitor/restart the job and move data back into the cloud when complete.

We use services for file and database storage, specifically AWS S3 and Dynamo respectively. Taken altogether, BrainLab CI scales incrementally with usage from a minimum of a single Webserver to run permanent services up to an arbitrary number of resources on demand.

### III.E Community Experiments and Data Controls

Data controls allow data providers to set guidelines and restrictions on how data are reused. Controls are defined in the context of an experiment, dictating what criteria data must meet to be included. For a continuous or incremental community experiment, this concept is well-defined; the experiment owner sets the guidelines.

We are still developing the model for data reuse and derived experiments. The challenge is to balance data controls from the data owner with exploratory data mining and ad-hoc reuse. At one extreme, we could prevent anyone from using contributed data without inheriting all controls. At the other, we could permit arbitrary reuse. Both are technically feasible. The specifics of sharing will depend on the application and domain and can be defined by the contributor.

We will implement and promote the concept of *compliant* and *non-compliant* data reuse, which permits ad-hoc usage, but also creates alerts for peer-review and auditing. Compliant reuse inherits all experimental controls. Non-compliant reuse allows derived experiments or data reuse to add or remove controls. In this way, there are no restrictions on exploratory data mining; anyone can use the data as they see fit. However, unrestricted reuse flags an experiment as non-compliant, alerting data owners and reviewers to inspect the manner of reuse carefully.

### III.F Broader Impacts

BrainLab CI will create a prototype that will lead to a national and then global experimental infrastructure for neuroscience with the goal of breaking down barriers to data sharing and reuse. The system eliminates objections to data sharing by allowing data contributors to implement controls on how data are used and audit/reproduce/rebut experiments run against their data. It also creates incentives to share data and use the system for data providers. Access to a large corpus of data will accelerate discovery and improve quality. Thousands of subjects will reveal statistically significant trends not apparent in the data available from a single laboratory. Larger studies that leverage community data will become commonplace and then evolve to standard practice in neuroscience.

BrainLab CI democratizes access to neuroscience data, enfranchising a new class of scientists that do not have the equipment, resources, or expertise to collect data. A larger, interdisciplinary community will expand the set of ideas brought to bear on neuroscience and diversifying the capabilities of the scientists that work on these data. Through tools like BrainLab CI, we can expand neuroscience beyond well-funded biomedical labs to a global community: to engage in brain science one needs a concept to explore and access to a Web browser. We expect new ideas and new approaches to arise from the social, cultural, and intellectual diversity of an international community.

BrainLab CI will also form the basis for a richer type of citizen science in which citizen scientists perform data exploration and discovery based on their own curiosity. This differs from existing approaches in which citizen scientists perform a specific, narrow task that contributes to a larger goal, such as galaxy spin and type classification in GalaxyZoo [5] and tracing of neurons in EyeWire [6]. The data controls provide guidance as to how one can appropriately combine data sets.

BrainLab CI proposes a novel approach to community neuroscience. It was conceived and feedback was taken from the neuroscience community during the NSF Global Brain Workshop. This initial design has garnered strong support from a network of collaborators (see Letters of Collaboration) that include top research institutes (Allen and HHMI Janelia), academic labs (UNM, Stanford), and medical researchers (JHU Medicine and Child Mind Institute). Additionally, we have taken feedback and will collaborate with European International efforts the Blue Brain Project and Human Brain Project (letter from Sean Hill).

## III.G Suitability for DARPA funding

The project leverages existing community resources as well as cloud computing, allowing us to deploy continuous integration without developing new software infrastructure. BrainLab CI builds upon the PIs' existing shared-infrastructure projects for storage, analysis, and pipeline processing of neuroscience data. Time-series data (electrophysiology), spatial data (MRI and atlases), and spatiotemporal data (optophysiology and functional MRI) will be stored and managed in the NeuroData project (http://neurodata.io), formerly known as the Open Connectome Project, which has been funded by an NSF/NIH CRCNS grant (1RO1EB016411-01), a BRAIN grant (1U01NS090449-01), and an NIH TRA (1R01NS092474-01). NeuroData includes built-in visualization and cloud data analysis based on Jupyter notebooks. MRI image processing, laboratory metadata, and registration will be provided by the NIH-funded MRI Cloud (http://mricloud.org), which uses NSF XSEDE computing (ACI-1053575) to provide scalable image processing to the open-science MRI community.

BrainLab CI will fulfill an immediate need for integration, sharing and analysis for neuroscience data. We have detailed two reference applications, MRI and neurophysiology. The approach is general and applies to directly to other imaging modalities that take large numbers of subject each with modest data size. These include MEG, EEG, ECoG, and PET. Adapting BrainLab CI to other disciplines will require minimal software development. The core software remains that same; the difference among disciplines lies in encoding domain expertise into, such as implementing data checks, processing pipelines, and visualizations and analysis for dashboards. These image modalities are used to understand and characterize a wide variety of neural disorders, including Alzheimer's, Parkinson's, Autism, ADHD, and TBI.

BrainLab CI builds upon two different, long-standing collaborations among computer scientists and neuroscientists, leveraging the tools built in both to create new capability. Neurodata has been run for five years by PIs Vogelstein and Burns and has developed tools for curated data sharing and annotation of MRI, electron microscopy [7], CLARITY [8], and Array Tomography [9] data. NeuroData is a registered Nature data repository and hosts definitive, archival data sets for annotation, visualization and analysis. MRICloud (http://mricloud.org), directed by Co-PI Miller, provides processing resources for the open-science MRI. It provide the image processing and shape analysis capabilities of MRI studio (also developed by Miller) through Web-services, fulfilling computation for the services using the NSF XSEDE funded supercomputers.

# IV Technical Plan

## IV.A Task 1: CLARITY

## IV.B Task 2: Multimodal MRI

## IV.C Task 3: Optophysiology

# V Statement of Work

## V.A Phase I

### V.A.1 Task 1: Mathematical Formalism

- **Goal:** *RAG Embedding:* Completion of the theoretical development and associated data structures of our RAG representation system. This includes establishing baseline methods for embeddings RAGs and populations thereof. Specifically, we will explore both JOFC and tensor factorization methodologies, to enable understanding of the computational and statistical advantages and disadvantages of each for embedding high-dimensional non-Euclidean RAGS.
- **Primary Site:** JHU
- **Milestone:** Demonstration that RAGs are able to meet TA1 goals, including encoding quantitative and qualitative knowledge, and express functional relationship among entities in complex systems.
- **Deliverables:** Description of mathematical framework and preliminary benchmarks evaluating performance on open access data sets and simulations.

### V.A.2 Task 2: Computational Infrastructure

- **Goal:** *Data Management:* Implementation of baseline algorithms for context-aware reasoning and inference using the representation. Development of an initial computational and data management platform, including establishing common data formats, common methods and format for query and analysis of results, and a common API through which all domain-specific users will access the framework. We will also extend our dense spatial and semantic databases, as well as our graph data format to support time-varying data, along with multimodal data.
- **Primary Site:** JHU
- **Milestone** Completion of Phase I prototype software and services.
- **Deliverables:** Open source software and documentation for end-to-end prototype.

### V.A.3 Task 3: Datafication

- **Goal:** *Data Ingest:* Completion of data ingest techniques. Completion of research and design into microscopic and mesoscopic specific analysis tools. Demonstration of auto-data ingestion and registration of multiple different modalities (functional to structural for both microscopic and mesoscopic data). More specifically, we will have ingested CLARITY, LFM, and M$^3$RI data into the same database schema; all M$^3$RI data will be co-registered.
- **Primary Site:** JHU
- **Milestone:** Demonstration of operational auto-ingestion and registration on two different use-cases.

- **Deliverables:** Open source software and documentation for datafication techniques, as well as image datasets ingested and RAGs estimated from all different data modalities.

### V.A.4 Task 4: Discovery

- **Goal:** *RAG Construction:* Completion of research and design into microscopic and mesoscopic specific analysis tools, by designing metrics appropriate for the different data modalities. This includes both the microscale and mesoscale functional time-series, converting into RAGs via utilizing qualitative information.
- **Primary Site:** JHU
- **Milestone:** Demonstration of RAG construction on both use cases.
- **Deliverables:** Open source software and documentation RAG construction techniques, as well as the derived RAGs available via our Web-services.

### V.A.5 Task 5: Program Management

- **Goal:** *Phase I:* Ensure successful execution of the effort. Manage the proposed effort using a proven methodology for project planning, resource allocation, task specification, and monitoring. Establish a baseline project plan with a list of tasks, specifications, requirements, and timelines; update plan periodically; document updates to the plan and share them with the project team and DARPA PM.
- **Primary Site:** JHU
- **Milestone** Meet Phase I goals.
- **Deliverables:** (1) Comprehensive quarterly technical reports that include updates systems architecture and progress made on milestones for Phase I; (2) Brief month reports, including preprints of technical reports; (3) Final Technical Report; (4) Monthly Financial Reports.

## V.B Phase II

### V.B.1 Task 1: Mathematical Formalism

- **Goal:** *FlashRAG:* Implementation of all embedding and construction methodologies in FlashGraph to enable scalable implementations and processing. Moreover, all constructed RAGs will obtain multilevel representations. We will build R bindings to enable easy use of FlashGraph for data scientists. We will check that our implementations and bindings yield approximately the same answer as benchmark methods, on data sufficiently small that benchmark methods can run.
- **Primary Site:** JHU
- **Milestone** Fully operational FlashGraph and R bindings for embedding and constructing methodologies.
- **Deliverables:** Open source software and documentation for end-to-end prototype for embedding and constructing RAGs. This includes an R package for FlashGraph.

### V.B.2 Task 2: Computational Infrastructure

- **Goal:** *Remote Access:* Implementation of prototype platform for remote access. This will include Web-services for uploading the raw data, and downloading the derived data products (RAGs and intermediate data products), as well as both 2D and 3D visualization and annotation tools, which will support multiple kinds of analytic overlays, all of which will support multiple data scales. Moreover, we will have made theoretical and practical refinements to the representation to enable scalable implementation of several foundational algorithms on

RAGs, implementing the embedding methodologies developed in Task 1 of Phase I into our semi-external memory formalism.
- **Primary Site:** JHU
- **Milestone** Fully operational Web-services supporting uploading, visualizing, annotation, querying, downloading, and analyzing the data.
- **Deliverables:** Open source software and documentation for end-to-end prototype. This includes an R package for FlashGraph which extends it capabilities to RAGs, rather than simply graphs.

### V.B.3 Task 3: Datafication

- **Goal:** *Data Register:* Integration of domain-specific computational models across modalities and scales. This includes completion of statistical multi-modal referencing, including alignment of structural and functional imaging data, for both microscopic and mesoscopic data sets. We will also complete functional inference capabilities. We will align data both via scaling up multidimensional out-of-core image alignment algorithms, and RAG matching, which extends graph matching by incorporating attributes. This will enable us to determine optimal alignments using data priors and known topological structure, rather than relying on images to align well.
- **Primary Site:** JHU
- **Milestone:** Demonstration of multi-modal registration for both microscopic and mesoscopic use cases.
- **Deliverables:** Open source software and documentation for datafication techniques, as well as registered multi-modal image datasets ingested and aligned RAGs from both microscale and mesoscale.

### V.B.4 Task 4: Discovery

- **Goal:** *RAG Summary Statistics:* Utilize RAG knowledge representation to estimate population moments, motifs, and/or modes from both micro- and meso-scale RAGs. More specifically, we will utilize the various joint embedding methodologies developed in Task 1 to estimate these summary statistics. The different approaches, JOFC versus tensor factorization, will enable incorporating different kinds of prior knowledge and constraints, so they will therefore lead to different bias/variance trade-offs. We will explore these options empirical on the real data, to complement our experiments in Task 1, to discover both (i) the best methods for estimation these summary statistics, and (ii) the best estimates of the summary statistics for the two different scales.
- **Primary Site:** JHU
- **Milestone:** Demonstration utility of RAG representation of data for estimating summary statistics for multi-modal data.
- **Deliverables:** Estimated summary statistics from micro- and meso-scale RAGs available for download in various formats, as well as open source code for the different estimators.

### V.B.5 Task 5: Program Management

- **Goal:** *Phase II Goals:* Ensure successful execution of the effort. Manage the proposed effort using a proven methodology for project planning, resource allocation, task specification, and monitoring. Establish a baseline project plan with a list of tasks, specifications, requirements, and timelines; update plan periodically; document updates to the plan and share them with the project team and DARPA PM.
- **Primary Site:** JHU

- **Milestone** Meet Phase II goals.
- **Deliverables:** (1) Comprehensive quarterly technical reports that include updates systems architecture and progress made on milestones for Phase II; (2) Brief month reports, including preprints of technical reports; (3) Final Technical Report; (4) Monthly Financial Reports.

## V.C Phase III

### V.C.1 Task 1: Mathematical Formalism

- **Goal:** *RAG Testing:* Demonstration of capabilities and objectives on both microscale and mesoscale data, as well as one additional SIMPLEX performer. To achieve this, we will extend our embedding methodologies, to derive provably approximately optimal embeddings for conducting one-sample and two-sample tests on RAGs; two fundamental testing procedures in statistics. Our tests will leverage our ability to efficiently sample RAGs, as they will be resampling based tests, analogs to the classic parametric and non-parametric bootstrap. We will apply these tests to test, for example, whether our data are sampled from relatively simple RAGs statistical models, and whether population means that we obtained in Phase I are significantly different from one another.
- **Primary Site:** JHU
- **Milestone:** Demonstrate our mechanism for relating qualitative and quantitative knowledge, and relating multiple heterogeneous datasets, on both heterogeneous scales (use cases). Specifically, testing whether multiple heterogeneous datasets are statistically different from one another.
- **Deliverables:** Description of capabilities, emphasizing generalizability to multiple use cases, open source code from implementing our tests, visualizations and numerical summaries of test results.

### V.C.2 Task 2: Computational Infrastructure

- **Goal:** *Local Analysis:* Completion of an integrated system on both heterogeneous scales (as well as additional domains). This system ingests and registers imaging data and semantic and qualitative knowledge, converts them into RAGs, allows query and recall, visualization, hypothesis generation, and analysis. We will also release open source packages containing all of the key resources, such as an R package (which calls igraph or FlashGraph) containing all of the developed methods, and GPU optimized visualization and annotation tools. This will enable anybody to implement analyses locally by running the code on their machine.
- **Primary Site:** JHU
- **Milestone** Fully operation Web-services to auto-ingest, register, store in compact representation, and query, as well as operate locally.
- **Deliverables:** Open source software and documentation for end-to-end prototype. including FlashGraphR and GPU optimized visualization and annotation tool.

### V.C.3 Task 3: Datafication

- **Goal:** *Quality Control:* Completion of quality control of all datasets. For each different modality, and each different scale, we will have already converted the raw data into RAGs. Now, we will build automatic quality controls, so that with each dataset, we automatically generate a quality control report, quantifying the key quality metrics appropriate for that data (see Table **??**).
- **Primary Site:** JHU
- **Milestone:** All datasets have been checked for quality.

- **Deliverables:** Open source software and documentation for datafication techniques, including quality control scripts, and resulting outputs.

### V.C.4 Task 4: Discovery

- **Goal:** *RAG Prediction:* Completion of toolset for analysis, modeling, and data-driven hypothesis generation and testing in both microscale and mesoscale heterogeneous use cases. This will include multiscale prediction; prediction of mouse status from microscale RAGs, and human personality from human RAGs.
- **Primary Site:** JHU
- **Milestone:** Successful integration with TA1 technology and end-of-program demonstrations of our integrated system on both microscale and mesocale.
- **Deliverables:** All data-derived products available for visualization utilizing our Web-services and quality control pages, as well as for download and further analysis using our open source code.

### V.C.5 Task 5: Program Management

- **Goal:** *Phase III:* Ensure successful execution of the effort. Manage the proposed effort using a proven methodology for project planning, resource allocation, task specification, and monitoring. Establish a baseline project plan with a list of tasks, specifications, requirements, and timelines; update plan periodically; document updates to the plan and share them with the project team and DARPA PM.
- **Primary Site:** JHU
- **Milestone** Meet Phase III goals.
- **Deliverables:** (1) Comprehensive quarterly technical reports that include updates systems architecture and progress made on milestones for Phase III; (2) Brief month reports, including preprints of technical reports; (3) Final Technical Report; (4) Monthly Financial Reports.

## VI Schedule and Milestones

Figure 4 provides the list of milestones organized by time and task. Please see §V for details.

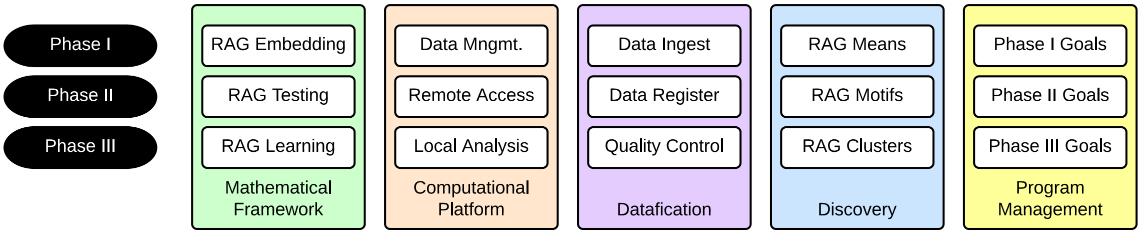


Figure 4: Schematic listing all milestones for this proposal. Black elements of this array denote quarters during which the work for that specific subtask will be completed. Deliverables will be delivered in the final quarter for the subtasks. Key milestone for each task in each phase correspond to the names of those tasks in §V, and §IV lists all subtasks. For Task 5 (bottom rows), the number of 'X's per element denotes the number of times the reports will be generated and sent.

# Literature Cited

[1] S. Sikka, J. T. Vogelstein, and M. P. Milham, “Towards Automated Analysis of Connectomes: The Configurable Pipeline for the Analysis of Connectomes (C-PAC),” in *Organization of Human Brain Mapping*, Neuroinformatics, 2012.

[2] S. M. Smith and et al., “Advances in functional and structural mr image analysis and implementation as fsl,” *Neuroimage*, vol. 23, no. Suppl 1:S, pp. 208–19, 2004.

[3] R. Burns and J. Vogelstein, “Neurodata: Enabling data-driven neuroscience at scale.” `http://neurodata.io`, 2016.

[4] R. S. Desikan, F. Ségonne, B. Fischl, B. T. Quinn, B. C. Dickerson, D. Blacker, R. L. Buckner, A. M. Dale, R. P. Maguire, B. T. Hyman, M. S. Albert, and R. J. Killiany, “An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest.,” *NeuroImage*, vol. 31, pp. 968–980, July 2006.

[5] K. Land, A. Slosar, C. Lintott, D. Andreescu, S. Bamford, P. Murray, R. Nichol, M. J. Raddick, K. Schawinski, A. Szalay, D. Thomas, and J. Vandenberg, “Galaxy Zoo: the large-scale spin statistics of spiral galaxies in the Sloan Digital Sky Survey,” *Monthly Notices of the Royal Astronomical Society*, vol. 388, pp. 1686–1692, Aug. 2008.

[6] J. S. Kim, M. J. Green, A. Zlateski, K. Lee, M. Richardson, S. C. Turaga, M. Purcaro, M. Balkam, A. R. B. F. Behabadi, M. Campos, W. Denk, H. S. Seung1, and the EyeWirers, “Spacetime wiring specificity supportsdirection selectivity in the retina,” *Nature*, vol. 509, no. doi:10.1038/nature13240, 2014.

[7] K. M. Harris, J. Spacek, M. E. Bell, P. H. Parker, L. F. Lindsey, A. D. Baden, J. T. Vogelstein, and R. Burns, “A resource from 3d electron microscopy of hippocampal neuropil for user training and tool development,” *Nature Scientific Data*, vol. 2, no. 150046, 2015.

[8] K. Chung and K. Deisseroth, “CLARITY for mapping the nervous system,” *Nature Methods*, vol. 10, pp. 508–513, May 2013.

[9] N. Weiler, F. Collman, J. Vogelstein, R. Burns, and S. Smith, “Synaptic molecular imaging in spared and deprived columns of mouse barrel cortex with array tomography,” *Nature Scientific Data*, vol. 1, no. 140046, 2014.