SANGAM: A Data Integration Framework for Studies of Stimulus-Circuitry-Gene Coupling in the Brain

M. Saxena[2], S. Kim[2], E. Alwagait[2], A.M. Khan[1], G.A. Burns[1], J. Su[3], A.G. Watts[1] and S. Ghandeharizadeh[2]*
Depts. of Biological Sciences[1] and Computer Sciences[2], University of Southern California, Los Angeles CA
Dept. of Computer Science[3], University of California, Santa Barbara, CA

571.3

Neuroscience Data Needs Integration!

The Challenge of Data Integration

"Assuming that neuroscientists agree to make their primary data available through well-structured web repositories, will that be sufficient to achieve the type of large-scale collaboration and data integration that we seek? As has been well acknowledged, placing data into shared data repositories is only a part of the battle; they must also be integrated into a body of cross-accessible knowledge, where results from disparate data sets can be accessed and understood on the basis of a common understanding of neural systems…" -- Martone ME et al. 2004 Nat Neurosci 7(5):467-472.

As the above quotation illustrates, neuroscientists are increasingly becoming aware of the growing need for computational tools to assist them with the integration of complex datasets. Here, we demonstrate the feasibility of Web Services in realizing such data integration for specialized sets of data that span multiple levels of analysis (molecular, tissue-level, behavioral/physiological).

The Solutions Offered by Web Services

What is a Web Service? A Web Service (WS) is a network-enabled application component with services-oriented architecture using standard interface description languages and communication protocols that facilitate easy development and deployment of data-intensive applications. Web Services operate much like web applications, where users enter some queries, submit them for processing, and receive displayed results on the web browser's interface. However, unlike web applications, they use standard XML representations to describe their inputs, outputs and available functions. XML representations allows WSs to be used as building blocks for Internet Database Management Systems, where different data sources can be integrated using an "integration execution framework".

Making WSs Useful: Shifting from asking "How" to asking "What"

Currently, WSs are not very useful for most neuroscientists because they require computer-programming skills. This requires scientists to think of HOW to access each different data source separately on the web, and HOW to write programs to utilize each Web Service available for such data sources.

Ideally, the scientist should simply state WHAT data he/she wants to see brought together from various Internet resources, and the system implements a data integration framework to a) identify the relevant online data sources and b) formulate an execution plan that manages the data flow between these sources.

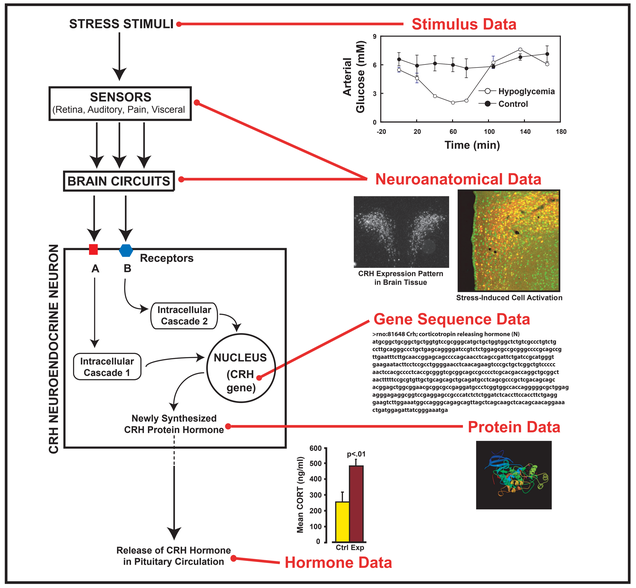
SANGAM: A New Way of Integrating Data

SANGAM is a data integration paradigm, designed to implement a "what-oriented" approach that empowers a neuroscientist to rapidly integrate and query functionalities of multiple biological data sources published as WS operations.

Specifically, SANGAM integrates information for neuroscientists specializing in studying gene and protein expression patterns in the brain that occur during various types of stressful stimuli. In this poster, we present the diversity of such data types, and how Sangam can bring them together by integrating the multiple WS operations that supply these data.

http://sangam.usc.edu

Diversity of Data



Data Sources

1- Kyoto Encyclopedia of Genes and Genomes (KEGG)
We use "KEGG Find" and "KEGG Get" functions provided by KEGG WS. "KEGG Find" finds all the matching molecules, their definitions and KEGG-specific IDs for a given term. "KEGG Get" takes the KEGG ID and finds the details of the molecule.

2- National Center for Biotechnology Information (NCBI)
We use NCBI's "eSearch", which takes a given term and criteria to find the IDs of all the matching molecules, and "eFetch", which takes a given molecule's ID and criteria to find its details.

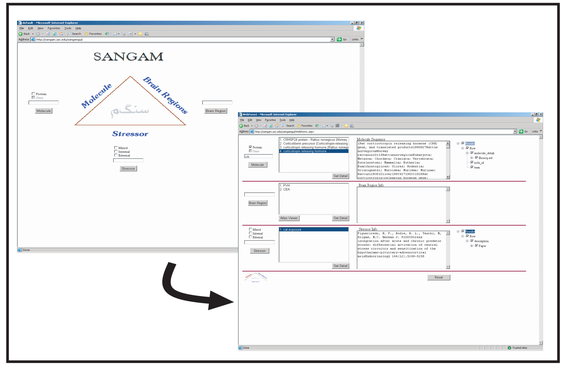
3- NeuroScholar (NS)
We use NS to map molecules to stressors and brain regions and to find brain region details.

4- NeuARt II
Once we are given a set of Brain Region IDs, we use NeuARt II to find atlas levels corresponding to each brain region and then extract details of each level to provide maps using the Swanson Brain Atlas.

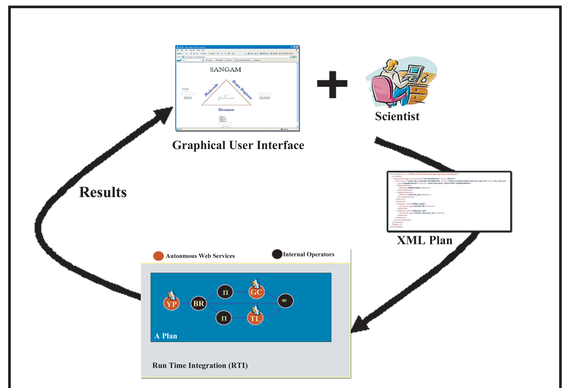
Sample Scenario

- The scientist requests the gene sequence for the CRH molecule and wants to find where this molecule is expressed in the brain.
- SANGAM composes a plan to invoke the KEGG1 and NCBI2 WSs (KEGG-find and NCBI eSearch) to retrieve the appropriate gene sequence.
- To narrow the results, the scientist may specify that she wants rat (*Rattus*) gene sequences for CRH
- SANGAM may then pursue two alternative approaches to realize this:
  1) It changes the invocation of the NCBI eSearch by providing it with '*Rattus*' condition
  2) It invokes eFetch WS with the ID for CRH to retrieve all matching molecules and employs SANGAM's select operator to filter out those corresponding to `*Rattus*'
- Finally, a scientist may want a Swanson Rat Atlas (SRA) display of brain regions for CRH. SANGAM identifies a lack of mapping between CRH and SRA, but offers one between CRH and the Paxinos and Watson atlas.
- The scientist now has two choices; both require interaction with the Semantic Data Management (SDM) component of the system:
  1) map CRH to the SRA by specifying the mapping between it and regions identified by the SRA;
  2) provide the mapping between the two atlases by specifying a relevant brain region (e.g., the medial nucleus of the amygdala, or "MEA" in SRA is roughly equal to "Me" in the Paxinos and Watson atlas).

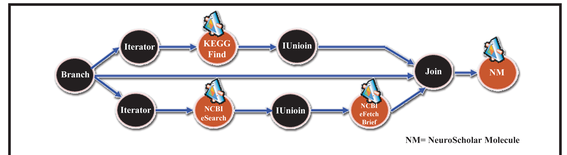
SANGAM GUI



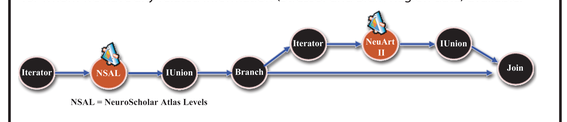
Components



Sample Queries



This plan takes a given term and find all the matching molecules (from KEGG and NCBI sources) with their definition and corresponding source-specific IDs but serve only those for whom we have any related information (Stressor and Brain Region data) available.



On a given set of Brain Region IDs, SANGAM uses NeuroScholar Atlas Level service to find the atlas levels corresponding to each brain region and then extract the details of each atlas level from the NeuARt II service to provide the atlas maps.

SANGAM Operators

Algebraic Operators

- ***Select :*** it consumes (1) an XML document, and (2) a selection clause that specifies selection conditions on elements of an XML document. It produces those XML objects that satisfy the selection predicate. The selection clause is a single predicate, a conjunctive or a disjunctive list of predicates. A predicate is a comparison between the value of two elements or an element and a constant. Note that the comparison operation might be one of {=, <, >, <=, >=, !=}.

- ***Project:*** it consumes (1) an XML document, and (2) a target list of elements. It produces a new XML document with objects consisting of the elements specified in the target list. Using the Project, the system can generate a document that consists of objects with only one element, MolName. Our current implementation of Project eliminates duplicate entries. Thus, if the projected element appears multiple times in the input XML document, it appears only once in the resulting XML document.

- ***Join:*** consumes two XML documents, and a Join clause specifying one element of each XML document that must satisfy a comparison operation. Note that the order of element names is not important due to our use of XML.
- ***Union*** consumes two XML documents with similar structure and merges them into one document, eliminating duplicate objects.

SANGAM Specific Operators

- ***Iterator :*** it consumes an XML document of objects and produces each object for consumption by the next WS one at a time. This is essential because a WS accepts only one XML formatted object as its input. Thus, if a WS is to be invoked with a set of XML objects, the Iterator operator is used to produce each object of the set one at a time. This operator can be implemented using either the push or the pull paradigm.

- ***Branch*** and ***Split :*** They signal a branching of the query tree. Each branch may execute independently of one another. However, at some point, these branches must reference a common operator (either a Join or a Union), synchronizing and merging them back into one branch. The key difference between Branch and Split is that Split is accompanied with selection clauses. To illustrate, recall the molecule document containing objects with elements: MolName and GeneSeq. With Branch, this document is duplicated and routed along each subsequent branch. With Split, each branch is provided with a selection clause, e.g., one branch is labeled with "MolName = CRH" and the other is labeled "MolName != CRH". In this case, Split constructs two documents: One with "MolName = CRH" and a second with other objects. It routes each document along the specified branch.

Future Research

Currently, we focus on techniques to enhance the response time of queries and smart eScience clients that empower a neuroscientist to tackle semantic (conceptual) heterogeneity of WS signatures and their content. Below, we describe each in turn.

Expediting Response Time
In order for SANGAM to be used on a daily basis by neuroscientists, it must provide a quick response time, defined as the amount of time elapsed from when a query is submitted to the time it completes execution. Our prior research in this area has included a middleware that decides when to compress the output of a WS in order to enhance response time. More recently, we have described the use of multiple replicas of a Web service in order to expedite the processing time of a query. This is most effective for those WSs that are updated rarely. We investigated the alternative scheduling and allocation policies. We showed that a policy that schedules WS replicas on-demand is better than one that schedules WS replicas when the query plan is initiated. We also showed that Least Response Time (LRT) policy performance is superior to other policies.

Smart eScience Clients
We believe that SANGAM's interface must be extended to capture the semantics of WSs regardless of the naming convention used in their signatures. This is similar to some of the work done by Phil Bernstein on schema equivalence. In particular, techniques that automate the equivalence of two WS signatures might be used to minimize the amount of repeated work performed by a neuroscientist. The smart eScience client enables two scientists to specify their assumptions, and SANGAM can initiate a dialogue to resolve some of the semantic heterogeneity found in any different, yet compatible WSs.

This Research is Sponsored By:

NATIONAL LIBRARY OF MEDICINE
NSF
Microsoft Research Labs
NATIONAL INSTITUTE OF NEUROLOGICAL DISORDERS AND STROKE
NIMH National Institute of Mental Health