# Web Platform Manual

Henry D. Priest, Ezra J. Umen, and Todd C. Mockler

June 12, 2015

## 1 Introduction

This document will provide a detailed description of each of the queries and result sets provided by the web front-end platform.

The web-based platform described herein is not a dataset specific entity. By utilizing the associated Java program to perform network analyses, and by utilizing the associated SQL import scripts, any network output from the algorithm can be loaded into an independently hosted web platform.

## 2 Platform, Terminology, and Structure

### 2.1 Feedback

Feedback from users is a critical part of any bioinformatic endeavor. Located at the top of every page is a large button titled "Report Issue/Make Suggestion". Clicking this takes the user to the web-based platform github repository.

New issues can be opened by clicking on the large, green, "New Issue" button in the upper right quadrant of the screen. On the following screen, a title must be entered, and the comment, suggestion, question, or bug report can be written in the provided area. A developer will reply to the issue so that a resolution can be obtained.

### 2.2 Terminology

A concrete understanding of terminology is critical for understanding the outputs of the web platform.

### 2.2.1 Abbreviations, Terms, and Definitions

**GCN**: Gene Co-expression Network

**Data Background**: A set of gene expression observations related to a set of gene or transcript identifiers.

**Node**: Computational object - represents a single gene and the associated gene-expression profile

**Target Node**: A node of-interest, usually because it is part of the input set of nodes provided by the user

**Edge**: A connection between Nodes. In the scope of this web-based platform, always refers to a sigmoid-transformed Pearson Correlation Coefficient

### 2.2.2 Graph Terminology

A **gene co-expression network** (GCN) is often thought of as a graph. A graph consists of a number of nodes and edges. Two nodes are connected by edges. In this application, the edge is "undirected" - it does not carry any other connotation other than the two nodes which are connected by the edge are related. As more and more nodes are added to a graph, and more and more edges which connect those nodes, the graph begins to take shape and meaning can be obtained.

An **Edge**, in this application, denotes a co-expression relationship. Edges represent an adjacency-transformed similarity value. Similarity values are derived from the pearson correlation coefficient, and are always between -1 and 1. The adjacency transform simply re-weights the similarity value so that low similarity values are given low weights, and high similarity values are given high weights. Therefore, adjacency values can be thought of exactly as you might think of a correlation. However, keep in mind that because of the transformed nature of the adjacency value, an adjacency value of 0.5 corresponds to a much higher raw correlation value, of about 0.9. In all cases, references to edge strength, edge value, or adjacency refer to the adjacency value, or in the case of plasticity networks, a difference between adjacency values.

A **node** represents a gene. In truth, a node represents a series of gene expression observations associated with a genomic locus annotated to represent a gene. Graph nodes in the web platform have several objects attached to them. Graph nodes take a place in the overall co-expression network. Graph nodes can have orthologs. Graph nodes are associated with a single expression data series per background. Graph nodes have an annotation that is common to all data backgrounds. Graph nodes are assigned to a single module per co-expression network, or possibly two modules in a plasticity network.

**Connectivity** (**K**) is a measure of how connected a node is in a particular network. Connectivity is a contextual measure - it can be computed based on a subgraph of a network, or on the total network. A node's connectivity is equal to the sum of all of the node's associated edge strengths within the given context. For example, modular K is the sum of all edges of a Target Node in which the non-Target Nodes of the edges are all within the same module as the Source Node. This is a powerful measure that has been shown to have biological meaning in gene co-expression networks. Genes with high connectivity within modules (module connectivity, or modular K), have been shown to play critical roles in the functions those modules are predicted to carry out. Therefore, the genes with the highest connectivity in a particular module are automatically of-interest if the module is of-interest. Network-level connectivity has not yet been shown to correspond to biological meaning, though it has been posited to be a signal that corresponds with criticality - in that loss of high connectivity genes on a network level will result in inviability of the organism.

**Modules** are groups of genes which are all interconnected. In this application, modules are non-overlapping, but for any given network, deterministic (the same data representing the same genes analyzed with the same settings will always produce an identical module set). However, this determinism should not be construed to imply correctness. Module membership is a binary state applied to a set of continuous variables. Put another way, a gene can be closely associated with 20 genes from module A. However, if that gene is also closely associated with 21 genes from Module B, it will be placed in Module B. This does not mean it is not closely and importantly associated with many genes from Module A. Thus, modular membership is an important guide to understanding predicted or implied function of genes. However, gene function and association is most accurately viewed as the overall set of edge-wise relationships.

## 2.3 Precomputed Metrics

In many cases, nodes will be displayed with pre-computed network metrics. These metrics have been shown to carry biological significance, or are expected to do so. The metrics are as follows, and each set relates to a single Target Node:

**Mean Expression (Mean Exp)**: The average within-dataset expression for the Target Node. This value is computed directly from the input expression values.

**Mean Expression Rank (Mean Exp Rank)**: The rank (highest first) of the Target Node's mean expression, in a list of all genes in the data background.

**K**: The connectivity of the Target Node in the entire GCN of the given data background. This is the sum of the edge strengths in the network which involve the Target Node.

**K Rank**: The rank (highest first) of the connectivity (K) of the Target Node, among all nodes in the given data background.

**Module**: The Module of which the Target Node is a member. For each GCN and data background, all nodes are members of zero or one Modules.

**Modular K**: The connectivity of the Target Node when only edges in which the non-Target Node is a member of the same module as the Target Node are included in the edge strength summation.

**Modular K Rank**: The rank (highest first) of a Target Node's Modular K, among all of the genes of the given Module. It has been shown that Nodes which are high ranking in Modular Connectivity often play critical roles in the function assigned to that module via functional term enrichment analysis.

**Modular Mean Expression Rank** (Modular Mean Exp Rank): The rank (highest first) of the mean expression of the Target Node, among nodes which are members of the same module as the Target Node. Utilization of this value, in conjunction with functional term enrichment analysis, has been shown to increase the hit rate of a targeted gene candidate screen by as much as 48-fold.

## 2.4 Identifiers

In general, the Identifiers that will need to be utilized in the web interface will be those that are utilized on the command line, during the network analysis process. In the case of plant species, these will often be the gene identifiers associated with the most recent genome release of the particular plant species on phytozome.gov.

A standard install of the database schema and web framework does not support automatic conversion from gene names (e.g. LHY), probeset IDs, or any other identifier to gene locus numbers.

## 2.5 Database Structure

The web platform which serves as the front end to the GCN analysis results resides upon a database. Although the innards of the database are not important to the biological interpretation of results, they are important towards understanding the functionality of the platform.

The database consists of one or more **data backgrounds**. Each background comprises the outputs of a single analysis. This is most likely the results of a network analysis performed on a single dataset (e.g., a single circadian time-course or stress treatment), or, a comparison via differential network analysis of a pair of such datasets.

Whenever a query is made, a data background is selected. All of the results displayed from this query derive from that data background, and no other. The nodes included in a data background are those that existed in the expression dataset analyzed by the GCN software. These node sets may not be common between individual data backgrounds, so there is no guarantee that a node representing your gene of interest is included in all data backgrounds present in a given instance of the web framework.

# 3 Network Queries

## 3.1 Gene Set Query

### 3.1.1 Use

The multi-gene, or gene set query allows identification of nodes which are related to a set of input gene identifiers. This set typically would involve a family of genes, of which the member-of-interest is unknown, or a set of genes involved in a metabolic pathway of interest. No limit is placed on the number of gene identifiers queried at any one time, but lists with more than ten members may produce very large result sets.

If the user selects the **"Include edges in which one node is in the Target Node list"** option, all edges in which one node is in the input list of gene identifiers will be selected for output.

If the user selects the **"Include edges in which both nodes are in the Target Node list"** option, all edges in which both nodes are in the input list of gene identifiers will be selected for output.

Each resultant Target Node/non-Target Node pair are displayed in a table. Target Nodes which have more than one Non-Target node are displayed more than once. Each non-Target Node is displayed along with the pre-computed metrics 2.3. Each gene locus identifier (both Target and Non-Target) in the displayed table is a click-able object that will display the available annotation information for that particular node.

### 3.1.2 Output Notes

All columns of the displayed Table are sortable, by clicking on the table header. The table can be searched for a specific gene identifier by utilization of the search box. Each column can be filtered by selection of the column from the filtering drop-down, and entering values into the Minimum and Maximum boxes directly to the right of the drop-down.

The present table view (including all nodes and edges represented by table rows on subsequent pages) can be converted into a graphical network view by clicking on the "Create Network Graph Based on Filtering Settings". No filtering is required to be applied to generate this graph. It is recommended that this feature it utilized with less than 1000 total nodes in the current table view.

The Target Node is listed in under the column labeled "Source". The Non-Target Node is listed under the column labeled "Gene". The number of connections each node possesses is listed under "Connections". The number of connections is not the same as the connectivity of the node under any circumstances.

## 3.2 Module Membership Query

### 3.2.1 Use

The Module Membership Query selects all nodes which are members of a given module.

### 3.2.2 Output Notes

Each node that is a member of the given module is displayed, along with their associated pre-computed metrics 2.3.

## 3.3 Gene Expression Query

### 3.3.1 Use

The Gene Expression Query accepts a list of gene identifiers present in the selected data background. The expression data series for each identifier which has expression data is returned.

### 3.3.2 Output Notes

The expression plot is highly customizable.

Expression data is filtered using a inner-quartile range (IQR) filter for outliers. Observations which lie more than 1.5 IQR from the median of a particular sample for a particular gene are discarded. The IQR value is the difference between the value of the 25th percentile and the 75th percentile.

Three plots are available - box-and-whisker plots, line-plots, and dot plots. Dot plots cannot be combined into single plots. Line plots can be combined into single plots, in which case the lines can be colored according to their gene, or

not colored to gain an overall impression of the expression of the group. Gene names below the line plot can be clicked to remove or re-add the expression data series corresponding to that gene to the plot.

Normalization is also highly customizable.

Gene expression data may be viewed as raw values, which can be difficult to interpret.

Mean normalization identifies the mean value of each gene expression series, and and divides the expression values of that gene by the the gene's mean value. This creates a data series in which the mean expression value is represented as "1.0" on the y-axis, and allows plotting of gene expression data series with large variation in absolute expression magnitude on the same plot.

Max normalization identifies the maximum value for each gene expression series, and divides each individual expression value by that maximum. This creates a data series in which the maximum value is 1.0, and all other data points represent a ratio of that maximum value.

Log-2 normalization simply transforms each expression value displayed by a base-2 Log function.

## 3.4 Identify Genes By Expression

### 3.4.1 Use

This query allows the identification of nodes which have expression profiles highly similar to a user-specified expression profile. The user specified profile is compared to the gene expression profile in a selected data background via the Pearson Correlation.

Once a data background is selected, a number of sliders are created, one to represent each individual sample. The user must then change the position of each slider to represent the desired expression profile. All sliders cannot be equal in value.

Once a desired profile is created, cutoffs must be selected.

Four cutoffs are required:

Minimum Correlation (Minimum R) - this is the value returned by the Pearson Correlation Coefficient. Genes whose correlations to the provided profile is less than this cutoff are excluded from the result set.

Number of Results - resultant "hit" genes are ranked by their correlation to the provided profile. Genes outside the top N hits (N provided by the user) are excluded from the result set.

Minimum Mean Expression - resultant hit genes with minimum mean expression less than this value are excluded from the result set.

Maximum Mean Expression - resultant hit genes with maximum mean expression above this value are excluded from the result set.

Once all four cutoffs are provided, the query is executed upon clicking of the "Submit" button.

#### 3.4.2 Notes

Great care should be taken when selecting the input expression profile. The Pearson Correlation functions by determining the mean of the two compared profiles. At each observation, if each data series observation falls on the same side of its respective mean (if point a from Series A is higher than mean of A, and if point b from Series B is higher than the mean of B), the correlation increases. If points fall on opposite sides of their respective means, the correlation decreases towards zero.

Thus, great care should be taken to ensure that the provided profile reflects the desired query. Non-variant regions should be exactly non-variant (in other words, all the same value), lest false-positive correlation be generated.

### 3.5 Query Orthologs

#### 3.5.1 Use

As an input to the orthology query, the user should supply one or more gene identifiers which exist in the annotation for the Given Species selection.

Identify the species within which you wish to find genes orthologous to your input set. Click on the "Submit" button.

For each gene which has an orthology relationship defined, one or more orthologous genes will be returned. This output is displayed in a table, with each row corresponding to a single pair of input gene and the obtained Ortholog.

The third field contains a link that allows querying the obtained Ortholog against its own network (if available). This will perform a Singe Gene Query, utilizing the Orthologous ID as input.

The Final field is a checkbox. Each checked gene can be utilized as the input to one of two queries. At the top of the page are a pair of buttons: "Network Query using Selected Genes", and "Expression Query using Selected Genes". Clicking one of these buttons sends the checked genes to the specified query as inputs.

### 3.5.2 Output Notes

The orthology query relies on the custom definition of a set of orthology relationships between genes of multiple species. Orthology in this case simply relates the genes of one species to another. Often this is done through sequence homology and synteny.

Of importance to biologists is that computationally derived orthology relationships are often wrong. computationally derived orthology relationships often do not account for experimentally determined functional orthologs between species. Genes can be identified as orthologous which are known to biologists to not be orthologs, or can be proven to not be orthologs by data not available to the computational system. If biological knowledge or data exists that disproves a predicted orthology relationship, of course, the biological evidence should take precedence over computational predictions.

Although the query is named an orthology query, the relationships defined need not be 1-to-1. A single source gene can have multiple genes designated to be 'orthologs' in a single target species, and a gene in the target species can serve as an ortholog for multiple genes in the 'source' species.

## 3.6 Query Annotations

# 4 Network Graph View

The visual representation of a Network Graph as a set of nodes connected by lines (edges) is a powerful method of quickly scanning a network to determine genes of interest.

To this end, on appropriate output pages, the set of nodes and edges displayed can be rendered as a network graph. The network graph may take time to render, depending on the number of nodes and edges included. The network graph is rendered utilizing the compute power of the user's computer. It is unwise to leave browser tabs which include a network graph open if the graph is no longer needed.

Each node is color-coded to represent the number of edges it has. This is distinct from the node connectivity, K. Nodes which served as the inputs to generate a list of edges for display are colored in dark blue. Each node is a click-able object which will display details, such as the gene name, annotation, pre-computed metrics (2.3), and orthology information, if any.