NTNU
Norwegian University of
Science and Technology

The BioGateway Cytoscape Plugin

A graphical interface to BioGateway within
Cytoscape

Stian Borgen Holmås

Master of Science in Computer Science
Submission date: December 2017
Supervisor: Keith Downing, IDI
Co-supervisor: Martin Kuiper, IBI

Norwegian University of Science and Technology
Department of Computer Science

Abstract

The BioGateway Knowledge Base, a semantic Systems Biology knowledge resource built using Semantic Web technologies, combines large datasets into an integrated repository for querying and information retrieval. It relies on a SPARQL-based web interface for interacting and retrieving data, but this remains challenging to use for biologists. This thesis presents the design, functionality and use cases of a tool developed by the author, that allows a more user-friendly way of interacting with BioGateway by enabling it to be queried from within the popular open-source network visualisation tool Cytoscape. By giving users access to the resources of BioGateway from within Cytoscape, this should make the knowledge stored in BioGateway more easily available to end-users.

Contents

List of Figures

CHAPTER 1

Introduction

In this chapter we will briefly introduce the concurrent evolution of high throughput genomics technologies, Systems Biology, the Semantic Web and the combination of the latter two into the field of Semantic Systems Biology.

1.1 Systems Biology

The field of Systems Biology is the study of biological systems as a whole, and not just their components. Biological systems consist of the interactions of cells, genes, proteins and other molecules that together make up a biological entity. By combining biological knowledge from multiple sources, biological models can be created on a computer, and these models can be used to run simulations and generate new hypotheses, which in turn can be tested in experiments[1][2]. Experimental data from such experiments will in turn give new insights that need to be reconciled with the model predictions, allowing the building of better models, thus completing the cycle.

However, a model is only as complete and good as the data it is based on. In order to build models accurate enough to formulate hypotheses which can be a basis for meaningful experiments, sufficient knowledge about the biological processes and interactions involved is important.

With the Human Genome Project's sequencing of our DNA[4] in the late 1990ies and early 2000s, we suddenly had access to the "raw code" behind our biological being. Together with the advances in computing power and high-throughput functional genomics technologies like gene expression microarrays[5], this data enabled researchers to model and identify the genes in the genome, the proteins they encode for and to some extent the function of these individual proteins. In 1999, IBM initiated its BlueGene supercomputer project to gain new biomedical knowledge through large-scale simulations of the interactions of these biological components[6], and it was at its completion in 2004 rated the fastest supercomputer in the world[7]. Another example would be the ambitious Folding@Home project, which was started in 2000 to crowdsource protein folding simulations to run in the background on participants' desktop PCs[8]. In 2006 the project had expanded to running on game consoles as well, and was the first computing system with over one petaFLOP (1 million billions calculations per second) of computing power in 2007[9]. It was clear that massive computational resources were put into biomolecular analysis.

While these high-throughput projects created a lot of new data about genes and

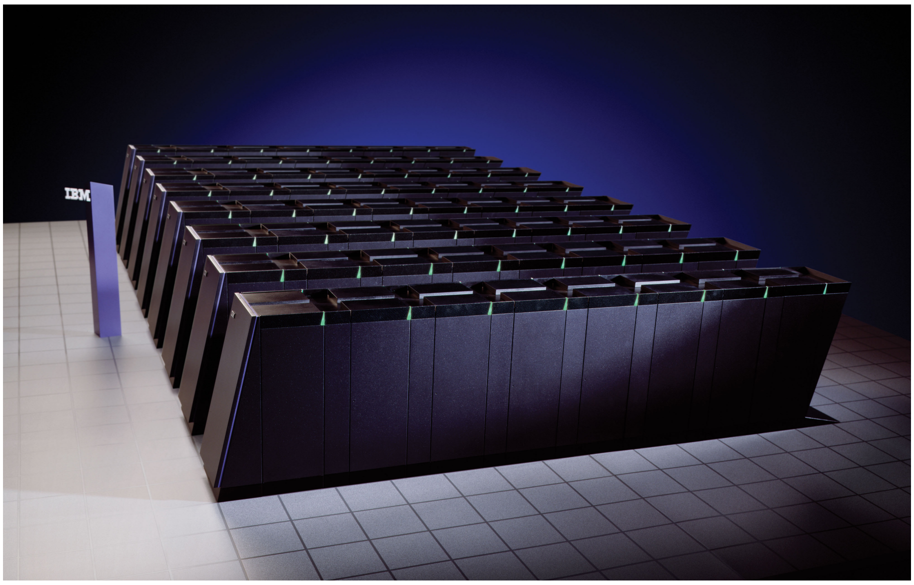
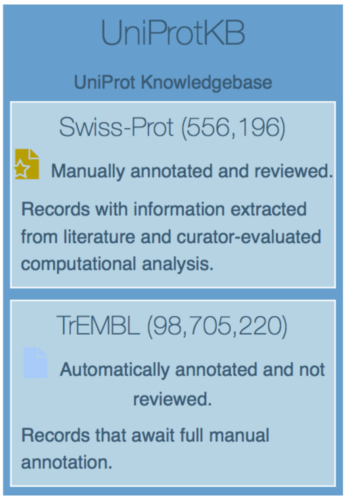


Figure 1.1: The BlueGene\L supercomputer[3].

the proteins they encode for, they would not necessarily translate into *knowledge* about how organisms work. The simple molecular models used in these simulations are well suited to model how genes and proteins behave in isolation, but not capable of modeling the result of the *interactions* between all these components. In order to explain higher-level biological processes like diseases, a more complex model of the biological systems is needed[1], and the field of research shifted from the study of the function of a gene or a protein to the study of function that is derived from interactions (a definition of Systems Biology phrased by Hans Westerhoff[10]). This school of thought gave rise to the field of Systems Biology.

1.1.1 The Data Revolution

The data from high-throughput sources are still an indispensable foundation for Systems Biology, and millions of data points from research and simulations were amassed in public databases like UniProt[11], IntAct[12] and many other databases. Figure 1.2 shows data available in UniProt, a prominent resource for protein and gene data, characterised as Swiss-Prot (human curated) and TrEMBL (computationally predicted). The difference between the size of human-curated and computationally generated data is striking, with the computationally-annotated TrEMBL database having almost 200 times more data, signifying the difference between high-throughput and manually curated data. While unverified computationally annotated



UniProtKB

UniProt Knowledgebase

Swiss-Prot (556,196)

Manually annotated and reviewed.

Records with information extracted from literature and curator-evaluated computational analysis.

TrEMBL (98,705,220)

Automatically annotated and not reviewed.

Records that await full manual annotation.

Figure 1.2: An overview of records available on the UniProtKB[11] website, totalling almost 100 million.

data might not be as trustworthy as that which has been reviewed, it cannot be ignored as a potential source of new knowledge, and tools would be needed to efficiently manage these huge datasets.

As databases and tools from different research communities popped up, it became increasingly hard to synchronise data between them. Simple things like gene and protein ID's was not even guaranteed to mean the same across databases, as there was no common framework. For instance, the *TP53* gene has the identifier "7157" in the NCBI Gene[13] datasets, while Ensembl[14] uses the identifier "ENSG00000141510.16".

Making sure that the data was successfully transferred from one dataset to another was left up to the researchers with their self-maintained parsers and spreadsheets of data.

1.1.2 Bioinformatics

Good computational tools are thus essential when working with such massive amounts of data and the analysis of complex biological systems, and the field of Systems Biology has therefore become heavily reliant on the field of Bioinformatics, which in its broadest form encompasses any use of computers for the management, archiving, processing, analysis and interpretation of data from the Life Sciences.

1.2 Semantic Web Technologies

> *In addition to the classic "Web of documents" W3C is helping to build a technology stack to support a "Web of data", the sort of data you find in databases. The ultimate goal of the Web of data is to enable computers to do more useful work and to develop systems that can support trusted interactions over the network. The term "Semantic Web" refers to W3C's vision of the Web of linked data.*
>
> – Description from the W3C consortium web page on Semantic Web[15]

The term "Semantic Web" is used to describe a set of technologies for linking relationships between data resources, turning it into "Linked Data" that a computer can perform meaningful processing on. Linked Data is a term used to describe the structuring of links in a network of data, in this case the Web. The "Web of Linked Data" adheres to these Semantic Web principles, and can be used to describe data as a network of relations between entities identified by their unique URIs[16]. The Semantic Web technology stack also includes formats and standards for storing and performing meaningful queries on such data.

Some of the core technologies are:

- A format to describe the relations between resources; the Resource Description Framework (RDF)
- A set of unique identifiers for every resource; the Universal Resource Identifiers (URIs)

- A query language for performing searches in this network of resources; SPARQL.

Semantic Web technologies provides powerful tools for the domain of Systems Biology thanks to the graph-like nature of the structure of Linked Data, and the efficient performance of graph-based queries. Large parts of the biological data domain can also be represented as networks, often called graphs, describing how genes, proteins and other (bio)molecules interact with each other, and how their interactions are connected in biological processes that enable cells and organisms. In the domain of bioinformatics, pathways are an example of graphs describing such interactions.

1.2.1 RDF Triples

The RDF format[17] is a standard for describing relations between resources on the Web. RDF uses triples of subject, predicate and object to describe the relations between entities in the Semantic Web. Compared to the hyperlinks on the normal Web, which only has a source (the webpage containing the link) and a destination (the link itself), RDF also captures the relation between the two in an explicit way that a computer can understand.

In bioinformatics terms, the relationship describing how a transcription factor protein (*STA5A_HUMAN*) controls a gene (*FOXP3*), can be described as a triple:

```
STA5A_HUMAN    molecularly controls    FOXP3
```

Listing 1.1: A specific relation in triple form

Using URIs, the RDF triple will be:

```
<http://identifiers.org/uniprot/P42229>   <http://purl.obolibrary.org/obo/RO_0002448>  <http://identifiers.org/ncbigene/50943>
```

Listing 1.2: RDF triple with URIs

Here, each of the URIs corresponds to a unique identifier describing each of the components in the triple. URIs are identified by being enclosed in < and > brackets. The RDF format also supports data that are not web resources. For instance, the RDF file can store the names of the object nodes as labels, which are enclosed in quotation marks.

```
<http://identifiers.org/uniprot/P42229>      <http://purl.obolibrary.org/obo/RO_0002448>  <http://identifiers.org/ncbigene/50943>
<http://identifiers.org/uniprot/P42229>      <http://www.w3.org/2004/02/skos/core#prefLabel>    "STA5A_HUMAN"
<http://purl.obolibrary.org/obo/RO_0002448>  <http://www.w3.org/2000/01/rdf-schema#label>       "molecularly controls"
<http://identifiers.org/ncbigene/50943>      <http://www.w3.org/2004/02/skos/core#prefLabel>    "FOXP3"
```

Listing 1.3: An RDF file containing 4 triples describing a relationship between the STA5A_HUMAN protein and the FOXP3 gene

The RDF example in listing 1.3 shows the RDF triples needed to describe this relation between this protein and gene, as well as the way the names of each entity is stored.

In the RDF format, the URIs are usually resolvable URLs, each of them linking to a specific resource specifying exactly what the component is. By pointing to human-readable resources, it is simple for humans to find the exact definition of these terms, while the unique identifiers also remove any ambiguity for a machine interpretation.

If the relationship had just been written in plain text, every researcher needs to agree on exactly what *"molecularly controls"* mean. By using an URI pointing to a resource describing it, the creators of the RDF resource implicitly agrees to the definition provided in the URI.

1.2.2 SPARQL

While RDF is a framework for storing data, the Semantic Web needs a toolset for querying such data, and SPARQL is the standard query language for the Semantic Web. The recursive acronym stands for SPARQL Protocol And RDF Query Language[18], and is a query language for reasoning on RDF resources. When executing a SPARQL query on a SPARQL server - called SPARQL endpoint - the result of a successful query is returned in the RDF format. An example of a very simple SPARQL query could be:

```
SELECT ?name
WHERE {
<http://identifiers.org/uniprot/P42229> skos:prefLabel ?name .
}
```

Listing 1.4: A simple SPARQL query fetching the label of a protein

In a similar manner to normal SQL, the *SELECT* keyword defines what data should be returned, and the *WHERE* statement defines the criteria to be matched. Within the brackets of the *WHERE* statement, a set of triples in RDF form are provided. In a query, the parts of a triple that start with a "?", are defined as variables. In this example, the part *"?name"* is the variable. The SPARQL server will attempt to find every possible part that matches the rest of the triples in the statement, and use them in the result. The example above would return any *"?name"* which has a *"skos:prefLabel"* relation from the entity with the URI *<http://identifiers.org/uniprot/P42229>*, in this case the preferred name of the protein.

This query also uses the RDF *skos* prefix syntax in the relation *skos:prefLabel.* Prefixes work as aliases, and the colon separates the prefix from the suffix. The *skos* prefix is a default prefix in some RDF stores such as Virtuoso[55], defined as *http://www.w3.org/2004/02/skos/core#.* Combining the prefix with the *prefLabel* suffix gives us the same relation as in listing 1.1, and this combination is done as part of the query evaluation.

A slightly more complex query would be to get the relationships like described in listing 1.1, but with the names of the entities instead of URIs.

```
SELECT ?protein ?relation ?gene
WHERE {
<http://identifiers.org/uniprot/P42229>
   <http://purl.obolibrary.org/obo/RO_0002448> ?geneURI .
<http://identifiers.org/uniprot/P42229> skos:prefLabel ?protein .
<http://purl.obolibrary.org/obo/RO_0002448> rdfs:label ?relation .
?geneURI skos:prefLabel ?gene .
}
```

Listing 1.5: A SPARQL query asking for the names of all genes molecularly regulated by a specific protein

In the query shown in listing 1.5, the *?geneURI* variable contains all values that have relation from the specified URI - *<http://identifiers.org/uniprot/P42229>* - of the type *<http://purl .obolibrary.org/obo/RO_0002448>*. The following 3 lines will map the *?protein* and *?relation* keywords to the labels of these entities. Finally, the *?gene* variable will be mapped to the labels of any matching *?geneURI*. If the RDF resource only contains the data in listing 1.3, the triple in listing 1.1 will be returned. If the RDF resource contains data on several genes molecularly controlled by the *STA5A_HUMAN* protein, all of these will be returned.

Graphs in SPARQL

An RDF resource can contain data stored in several sub-graphs. Limiting a search to a specific graph can limit the scope of a query and reduce the amount of triples needed to be evaluated in the query. In a database aggregating datasets from several different resources, each graph might represent the origin dataset of the data, enabling users to search for data originating from a specific source.

1.3 Semantic Web and Systems Biology

The rise of high throughput functional genomics technologies together with the increased capacity to computationally analyse these data has given rise to an overwhelming number of databases covering virtually all aspects of biological systems[19]. However, in practice it is not trivial to recover and integrate information from various resources into one conceptual model of a biological system, which is the first step of Systems Biology. Many resources have their own idiosyncratic internal standard, use of IDs, XML data exchange format, or use of gene and protein synonyms. Many of these issues with the vast, unmanageable databases can be addressed using Semantic Web technologies. The global scope of URIs would for instance ensure that biological entities and relations have one unambiguous name. The networked properties of the Semantic Web makes it possible to find relations across datasets, and the SPARQL language enables researchers to query for such relations.

Several initiatives to merge Semantic Web technologies with the Life Sciences has been proposed[20][21], and the Semantic Web Health Care and Life Sciences Interest Group (HCLSIG)[22] was set up within the W3C consortium - the proponents behind the Semantic Web technologies. While publications in the Life Sciences related to the Semantic Web remains relatively low compared to Systems Biology (shown in

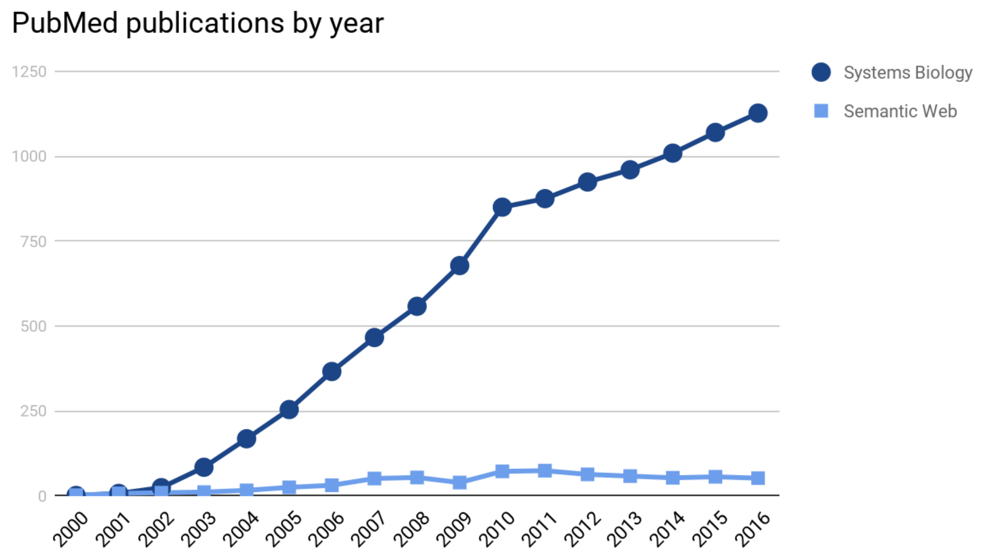


Figure 1.3: Yearly number of PubMed articles mentioning "Systems Biology" or "Semantic Web" in the title or abstract.

figure 1.3), there has been an increase in publications regarding Semantic Web, indicating insterest in the Semantic Web within the Life Sciences.

1.3.1 Semantic Systems Biology

Antezana et. al have introduced the term *Semantic Systems Biology* to describe the possibilities of merging Systems Biology and the Semantic Web field[23][24]. In their 2013 paper[25] they explain the "Semantic Systems Biology Cycle" as a modification of Kitano's cyclical view of Systems Biology[1]. Where Kitano describes Systems Biology as a cyclical iteration of computational modelling and simulation with a wet-lab laboratory phase, in Semantic Systems Biology the computational modelling and simulation is replaced by querying and reasoning.

The Semantic Systems Biology Cycle contains these phases:

1. All available information and knowledge is converted to one Semantic Web format

2. Queries or reasoning tasks are performed on this resource, and new inferences or assertions can be made

3. Based on this new hypotheses are formulated

4. Hypotheses are verified in wet-lab experiments

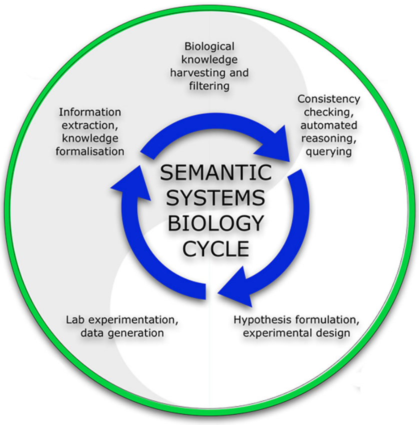

Biological knowledge harvesting and filtering
Consistency checking, automated reasoning, querying
Information extraction, knowledge formalisation
SEMANTIC SYSTEMS BIOLOGY CYCLE
Lab experimentation, data generation
Hypothesis formulation, experimental design

Figure 1.4: The Semantic Systems Biology cycle.

5. Information from these new experiments is curated and integrated into the knowledge base

The augmented knowledge base leads to better query results and reasoning, and the cycle can be repeated to formulate new hypotheses, perform new experiments, and gain new insights.

Two of the pioneering resources developed within the framework of Semantic Systems Biology include Bio2RDF and BioGateway. The former attempts to be as comprehensive as possible while the latter focuses on higher quality data.

1.3.2 Bio2RDF

First proposed in 2008, Bio2RDF is an initiative labeling itself as a "mashup system" which combines several public bioinformatics databases such as UniProt[11], Gene Ontology[26], Entrez Gene[27] and KEGG[28] into one unified RDF resource[29]. Using the Semantic Web principle of unified URIs and common ontologies, it makes it possible to search across data from multiple databases at the same time, without the need to translate identifiers between them. Bio2RDF also provides a public SPARQL endpoint[30], enabling 3rd party services to access its unified repository for export, visualisation and further analysis.

1.3.3 Origins of BioGateway

The BioGateway resource was initiated as a proof-of-concept of integrating Semantic Web technologies with Life Sciences knowledge bases, initiated by the Kuiper Lab[31] in 2008[23]. Together with Bio2RDF[29], it was one of the most comprehensive Semantic Web resources for biomedical knowledge at the time, and it allowed new ways of reasoning with the data from the various sources it had unified. By giving the entities in the database semantic meanings and unifying their identifiers, it was now possible to perform searches across different types of data to find the specific subset of interest. Powerful as these resources may be, their use by the biological research community has been less than anticipated, mostly because of the lack of a user-friendly interfaces.

CHAPTER 2

Motivation and Related Work

2.1 Common Systems Biology Resources

Several resources with biomedical data are available online, and are commonly used by systems biologists to build and refine their biomedical model networks. Most resources are used through web-based interfaces where users can enter keywords to search through the databases, and some of them also provide SPARQL endpoints for more advanced queries.

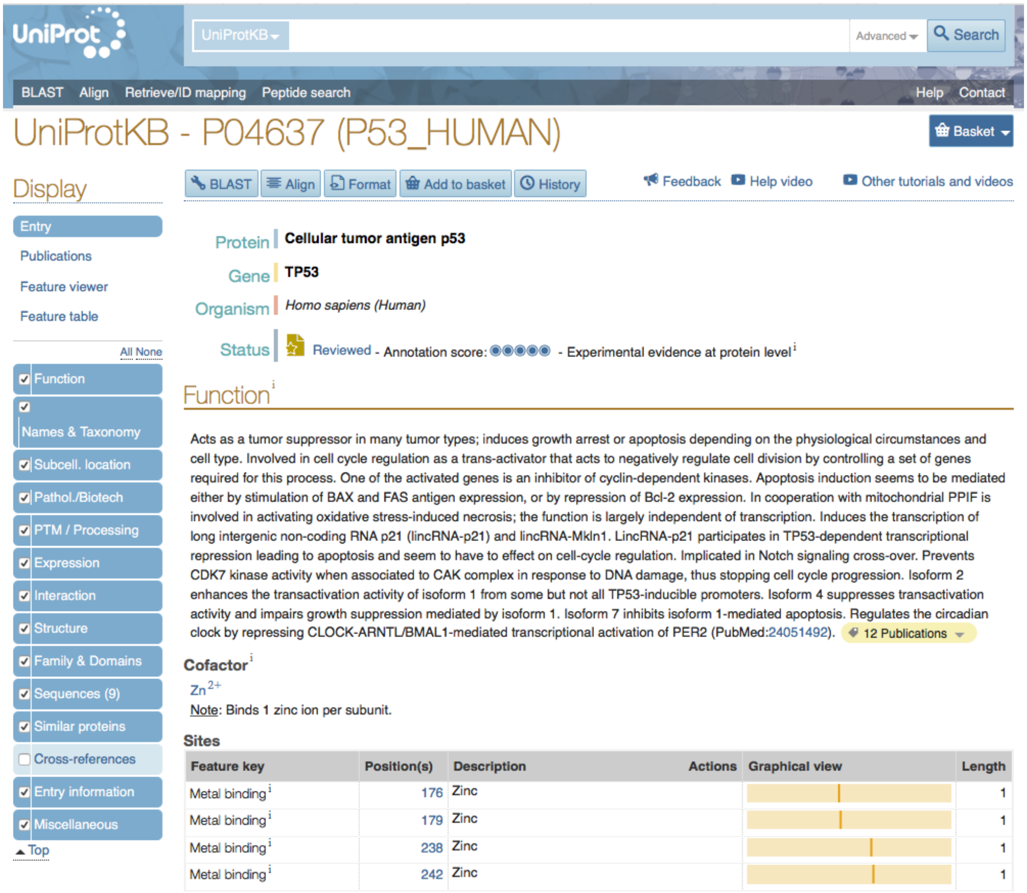
2.1.1 AmiGO

AmiGo[32] is a web service tool provided by the Gene Ontology Consortium to search through the Gene Ontology Annotation (GOA) dataset. Curators around the world who sift through genomic and proteomic data use the definitions and GO terms provided by GO to annotate or curate the genes and proteins in their favorite species. The GOA project aims to provide high-quality Gene Ontology (GO) annotations to proteins in the UniProt Knowledgebase (see section 2.1.2), but the annotations and ontology terms are also queryable in AmiGO. GOA is responsible for the integration and release of GO annotations to many "proteomes". Because of the multi-species nature of the UniProtKB, GOA also assists in the curation of another 200,000 species. This involves electronic annotation and the integration of high-quality manual GO annotation from all GO Consortium model organism groups and specialist groups. This effort ensures that the GOA dataset remain a key reference and a comprehensive source of GO annotation for all species.

2.1.2 UniProt

The UniProt knowledge base is a comprehensive resource containing almost 100 million entries in its UniProtKB knowledge base[11]. It is searchable through its web interface, and the result pages, shown in figure 2.1, combine relevant information from several sources such as GO Annotations from GOA[26], protein-protein interactions (PPIs) form IntAct[12], associated diseases and more.

UniProt is also accessible through its SPARQL endpoint for users comfortable with SPARQL[35], and the results are downloadable in XML, JSON and CSV.



UniProtKB - P04637 (P53_HUMAN)

Protein Cellular tumor antigen p53

Gene TP53

Organism *Homo sapiens (Human)*

Status Reviewed - Annotation score: ●●●●● - Experimental evidence at protein level

Function

Acts as a tumor suppressor in many tumor types; induces growth arrest or apoptosis depending on the physiological circumstances and cell type. Involved in cell cycle regulation as a trans-activator that acts to negatively regulate cell division by controlling a set of genes required for this process. One of the activated genes is an inhibitor of cyclin-dependent kinases. Apoptosis induction seems to be mediated either by stimulation of BAX and FAS antigen expression, or by repression of Bcl-2 expression. In cooperation with mitochondrial PPIF is involved in activating oxidative stress-induced necrosis; the function is largely independent of transcription. Induces the transcription of long intergenic non-coding RNA p21 (lincRNA-p21) and lincRNA-Mkln1. LincRNA-p21 participates in TP53-dependent transcriptional repression leading to apoptosis and seem to have to effect on cell-cycle regulation. Implicated in Notch signaling cross-over. Prevents CDK7 kinase activity when associated to CAK complex in response to DNA damage, thus stopping cell cycle progression. Isoform 2 enhances the transactivation activity of isoform 1 from some but not all TP53-inducible promoters. Isoform 4 suppresses transactivation activity and impairs growth suppression mediated by isoform 1. Isoform 7 inhibits isoform 1-mediated apoptosis. Regulates the circadian clock by repressing CLOCK-ARNTL/BMAL1-mediated transcriptional activation of PER2 (PubMed:24051492).

Cofactor

Zn^{2+}

Note: Binds 1 zinc ion per subunit.

Sites

Feature key	Position(s)	Description	Actions	Graphical view	Length
Metal binding	176	Zinc			1
Metal binding	179	Zinc			1
Metal binding	238	Zinc			1
Metal binding	242	Zinc			1

Figure 2.1: Information listed on the *P53_HUMAN* protein in UniProt[33].

2.1.3 IntAct

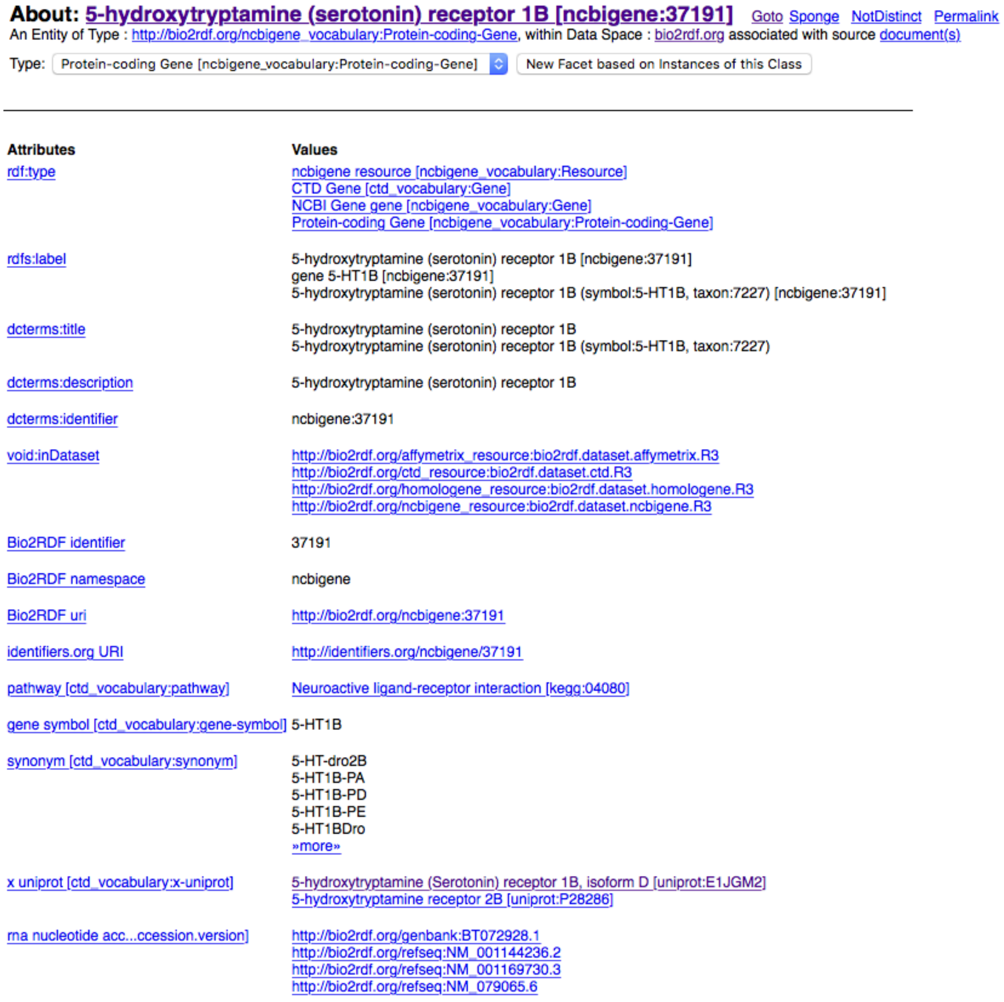
IntAct[12] is a public dataset for PPI data, and dataset can be downloaded as XML or in a raw text format, but at a massive 3.16 GB this file is hardly suitable to be opened in a spreadsheet. Thankfully the website also provides a web resource for searching through its vast amounts of data which includes experimental sources for the annotated interactions.

2.2 Bio2RDF

This resource was mentioned in section 1.3, and is among the most comprehensive bioinformatics RDF resources with a collection of 11 billion triples in 35 datasets[36]. It has a SPARQL endpoint which can be used for querying, and a simple web interface for looking up information on an entity, providing a comprehensive list of relations to other biological entities.

2.3 Cytoscape

Cytoscape is a popular multi-platform open-source network visualisation tool built for visualising and analysing with biological systems. It supports many common



About: 5-hydroxytryptamine (serotonin) receptor 1B [ncbigene:37191] Goto Sponge NotDistinct Permalink

An Entity of Type : http://bio2rdf.org/ncbigene_vocabulary:Protein-coding-Gene, within Data Space : bio2rdf.org associated with source document(s)

Type: Protein-coding Gene [ncbigene_vocabulary:Protein-coding-Gene] New Facet based on Instances of this Class

Attributes	Values
rdf:type	ncbigene resource [ncbigene_vocabulary:Resource] CTD Gene [ctd_vocabulary:Gene] NCBI Gene gene [ncbigene_vocabulary:Gene] Protein-coding Gene [ncbigene_vocabulary:Protein-coding-Gene]
rdfs:label	5-hydroxytryptamine (serotonin) receptor 1B [ncbigene:37191] gene 5-HT1B [ncbigene:37191] 5-hydroxytryptamine (serotonin) receptor 1B (symbol:5-HT1B, taxon:7227) [ncbigene:37191]
dcterms:title	5-hydroxytryptamine (serotonin) receptor 1B 5-hydroxytryptamine (serotonin) receptor 1B (symbol:5-HT1B, taxon:7227)
dcterms:description	5-hydroxytryptamine (serotonin) receptor 1B
dcterms:identifier	ncbigene:37191
void:inDataset	http://bio2rdf.org/affymetrix_resource:bio2rdf.dataset.affymetrix.R3 http://bio2rdf.org/ctd_resource:bio2rdf.dataset.ctd.R3 http://bio2rdf.org/homologene_resource:bio2rdf.dataset.homologene.R3 http://bio2rdf.org/ncbigene_resource:bio2rdf.dataset.ncbigene.R3
Bio2RDF identifier	37191
Bio2RDF namespace	ncbigene
Bio2RDF uri	http://bio2rdf.org/ncbigene:37191
identifiers.org URI	http://identifiers.org/ncbigene/37191
pathway [ctd_vocabulary:pathway]	Neuroactive ligand-receptor interaction [kegg:04080]
gene symbol [ctd_vocabulary:gene-symbol]	5-HT1B
synonym [ctd_vocabulary:synonym]	5-HT-dro2B 5-HT1B-PA 5-HT1B-PD 5-HT1B-PE 5-HT1BDro »more»
x uniprot [ctd_vocabulary:x-uniprot]	5-hydroxytryptamine (Serotonin) receptor 1B, isoform D [uniprot:E1JGM2] 5-hydroxytryptamine receptor 2B [uniprot:P28286]
rna nucleotide acc...ccession.version]	http://bio2rdf.org/genbank:BT072928.1 http://bio2rdf.org/refseq:NM_001144236.2 http://bio2rdf.org/refseq:NM_001169730.3 http://bio2rdf.org/refseq:NM_079065.6

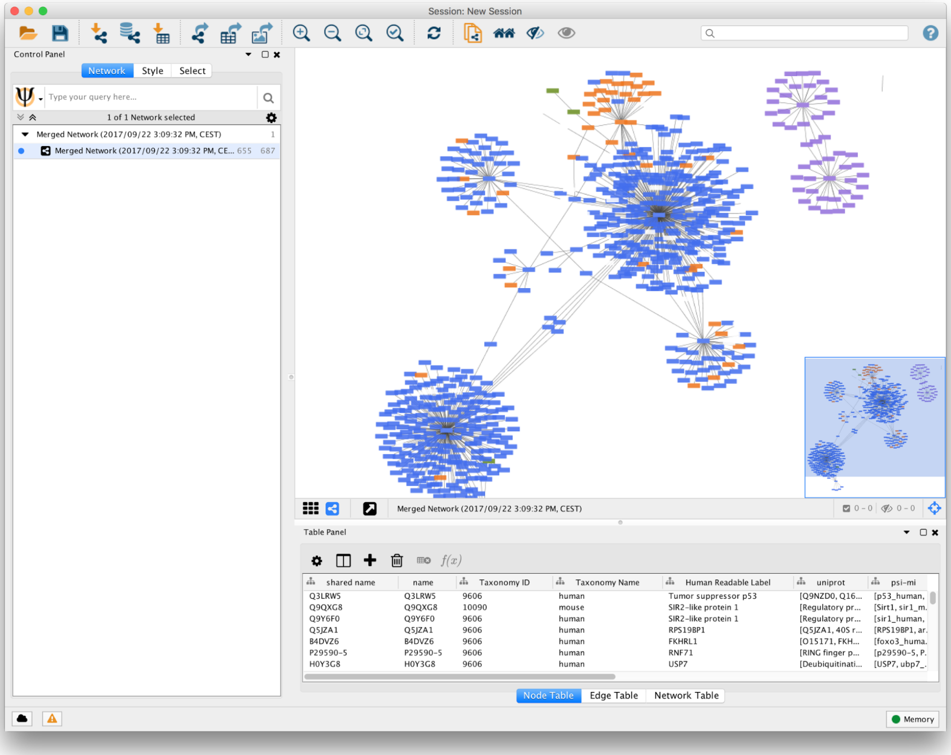
Figure 2.2: Information page about the *5-HT1B* gene in Bio2RDF[37].

file formats such as a simple interaction file (SIF), and several XML-based formats (GML, BioPAX, KGML, SBML, OBO)[38].

The application also has support for importing data from several well-known Life Science databases, and comes with a large set of tools for manipulating and visualising large networks of nodes and edges, including an extensive set of layout algorithms including the yFiles layout algorithms[39], and a very customisable visual style manager[40].

Cytoscape is built as a core application, and is intended to be extended by 3rd-party plugins[41]. The current version of Cytoscape, Cytoscape 3, is referring to these plug-ins as “Apps”, and includes its own “Cytoscape App Store”[42] where these plug-ins can be shared with the Cytoscape user community. These apps expand the functionality of the Cytoscape platform, and can address the needs of a particular user group (for instance a specific type of analysis) by enabling features not otherwise available in the core application.

As an open-source Java project, the source code for the entire Cytoscape application is available at http://github.com/cytoscape. Open-source also means free (as in both beer and speech), making Cytoscape a popular choice in the Systems Biology community. The large open-source community developing and providing



shared name	name	Taxonomy ID	Taxonomy Name	Human Readable Label	uniprot	psi-mi
Q3LRW5	Q3LRW5	9606	human	Tumor suppressor p53	[Q9NZD0, Q16...	[p53_human,
Q9QXG8	Q9QXG8	10090	mouse	SIR2-like protein 1	[Regulatory pr...	[Sirt1, sir1_m.
Q9Y6F0	Q9Y6F0	9606	human	SIR2-like protein 1	[Regulatory pr...	[sir1_human,
Q5JZA1	Q5JZA1	9606	human	RPS19BP1	[Q5JZA1, 40S r...	[RPS19BP1, ar.
B4DVZ6	B4DVZ6	9606	human	FKHRL1	[O15171, FKH...	[foxo3_huma.
P29590-5	P29590-5	9606	human	RNF71	[RING finger p...	[p29590-5, P.
H0Y3G8	H0Y3G8	9606	human	USP7	[Deubiquitinati...	[USP7, ubp7_.

Figure 2.3: The Cytoscape application.

feedback to Cytoscape ensures that the platform can continue to evolve to maintain its broad use and relevance in a changing field.

2.3.1 Pros and cons of Cytoscape as a visualisation platform

Advantages

The most immediate advantage of using an established platform such as Cytoscape, is that not all features needs to be built from scratch. Functionality such as layout algorithms and visual style management could take a lot of time to develop, even if 3rd-party open-source libraries were used. Increasing the scope of development to such a large project would involve many unknowns, and maintaining such a large app in the future would require more work.

Cytoscape's large user base and "App Store" also provides a valuable platform for getting our application out to the relevant users. Users familiar with Cytoscape's toolsets will not need to re-learn this functionality in a new application. Also, by sharing the common core application and data models in Cytoscape, an app can be used alongside other apps to build upon the same network models.

The Cytoscape app development community also has some documentation on how to develop 3rd-party apps[43]. And as the core application is open-source, it is possible to read the core source code to figure out where errors arise from during development.

Challenges

Developing an app in the Cytoscape environment requires knowledge about how Cytoscape works and communicates with its plug-ins. Because the app is being executed by Cytoscape instead of as an standalone application, the app will need to play along with several other components, such as the visual manager. Synchronising events across several components is not always trivial, and can lead to bugs that are hard to resolve. Depending on other components to take care of layout, visualisation and data models will limit an app's functionality to what these components provide.

2.4 Other Cytoscape RDF plugins

2.4.1 RDFScape

As one of the first Cytoscape plugins to offer RDF querying support, RDFScape was developed by Andrea Splendiani as a prototype proof-of-concept in 2007[44]. The plugin featured several ways of querying RDF resources, including a SPARQL editor, "visual queries" and contextual right-click menus.

Unfortunately, the latest version of the plugin, 0.4.1 is from 2008[45], and not available for Cytoscape 3. The rest of the website does not contain any news since 2010, and the download links are dead. It is reasonable to believe that the plugin has now been abandoned.

2.4.2 General SPARQL

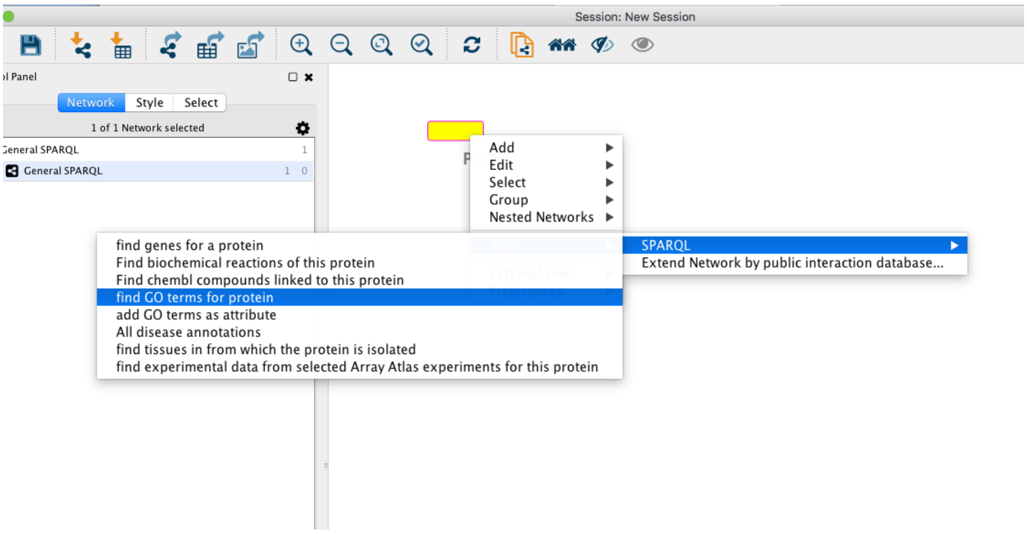

Session: New Session
l Panel
Network
Style
Select
1 of 1 Network selected
General SPARQL 1
General SPARQL 1 0
Add
Edit
Select
Group
Nested Networks
SPARQL
Extend Network by public interaction database...
find genes for a protein
Find biochemical reactions of this protein
Find chembl compounds linked to this protein
find GO terms for protein
add GO terms as attribute
All disease annotations
find tissues in from which the protein is isolated
find experimental data from selected Array Atlas experiments for this protein

Figure 2.4: Searching for GO terms inside Cytoscape with the General SPARQL plugin.

General SPARQL has been developed as an open-source Cytoscape app by the bioinformatics consulting company General Bioinformatics[46], and was released in 2016[47]. It enables users to query various SPARQL endpoints such as UniProt and Bio2RDF from within Cytoscape. It includes an interface to customise the SPARQL queries available in contextual menus, and result data is automatically imported directly into the current Cytoscape network.

However, at the time of writing, the default configuration of the app did not work correctly. After some research it turned out to be a small error in the plugin's configuration XML file, where the URI for UniProt's taxonomy graph was set to http://sparql.uniprot.org/taxonomy/, instead of http://sparql.uniprot.org/taxonomy. The extra "/" at the end broke the default queries, causing them to return no data. Looking at GitHub repository of the app[48], the default config file has not been updated since 2015. Unless UniProt changed their taxonomy graph URI very recently, it is reasonable to suspect that this plugin is no longer regularly maintained. The plugin will work if the bug is corrected, and enables the user to get contextual menu actions of relevant queries when right-clicking a node, as shown in figure 2.4.

2.5 The BioGateway Database

BioGateway is a semantic graph database, also called "triple-store", in which the data is stored as networks with nodes and edges, as opposed to the common relational databases where the data is stored in tables, like in a spreadsheet. The database relies on semantic web technologies to aggregate several large databases into one searchable data resource. Due to the nature of graph databases, cross-searching between arbitrary datasets does not come with an extra computational cost, such as with relational databases. This allows users to query for data matching criteria across several datasets in a single step. BioGateway also adheres to the semantic web principle of giving all entities a specific unique identifier URI, describing entities such as genes or proteins, and generally points to a human-readable web resource describing that entity. The same URIs are used across all datasets, so the user can use the same URI to query data about an entity from all the datasets without the need to manually merge the data afterwards.

2.5.1 Data Contained in BioGateway

BioGateway currently contains data from:

- OBO Foundry Ontologies
- GOA Associations
- UniProt
- IntAct
- NCBI Gene and RefSeqGene
- NCBI Taxonomy
- TF-TG data (beta)

The database also contains datasets of pre-computed relations inferred from the semantics implemented in the resources for quicker searches. For instance, if a user wants to limit a data set of proteins to the ones that inhere in mammals, the database has already inferred that all proteins that inhere in mice, humans, etc., also inheres in mammals. Thus it is possible to search for entities in mammals directly without resolving which taxa are mammals first.

Gene Ontology

The OBO Foundry Ontologies[49] is a collection of ontologies for the Life Sciences community. One of the goals of the OBO Foundry is to provide standardised URIs for biological entities under their own obofoundry.org namespace. These kind of URIs are essential to use semantic web technologies to store information about these biological entities.

The by far most important of these ontologies is the Gene Ontology (GO), an initiative aiming to develop a common set of terms for biological processes, molecular functions and cellular components of all organisms. The terms are hierarchically organised in directed acyclic graphs (DAGs), or "trees". The three root nodes are *biological process*, *molecular function* and *molecular function*, with sub-terms being more specific. For instance, the GO term *response to ionizing radiation* is an ancestor of the term *response to radiation* and ultimately the root term *biological process.* It also has child terms such as *response to x-ray* and *response to gamma radiation.* Researches can use these terms to describe their research findings in an accurate, standardised way, which also leaves room for describing the inherent uncertainty in some biological results - i.e. if it is not clear from the results if something was a response to x-ray or gamma radiation.

IntAct, UniProt and Gene Ontology Annotations

These datasets are based on the same data as described in section 2.1.1, 2.1.2, and 2.1.3, for GOA, UniProt, and IntAct respectively. BioGateway uses the UniProt protein identifiers as a basis for its protein URIs.

NCBI Taxonomy

> *"The Taxonomy Database is a curated classification and nomenclature for all of the organisms in the public sequence databases. This currently represents about 10% of the described species of life on the planet."*
>
> – From the NCBI Taxonomy Database website[50]

The NCBI Taxonomy ontology is used to classify genes and proteins into their associated species, and allows queries to limit searches to specific organisms.

NCBI Gene

NCBI Gene[13] is a set of gene identificators, created in an effort to provide unique identifiers to genes so that they can be referenced across different datasets and databases. BioGateway uses the NCBI Gene identifiers as basis for the URIs for all its genes.

NCBI RefSeqGene

The NCBI RefSeqGene[51] project is a subset of NCBI RefSeq, and contains data on gene labels, protein encoding and organism association.

Transcription Factor - Target Gene interactions (beta)

For regulatory pathway building, data about transcription factors regulating specific target genes is important. Such data exists in a number of forms: as curated knowledge like PAZAR[52], TFactS[53], TRRUST[54] and others (to be added to BioGateway in the near future), or as information obtained by text mining. The Kuiper and Lagreid groups at NTNU have been collaborating with the group of Valencia (now at the Barcelona Supercomputing Centre, Spain) on the development of text corpora and classifiers to retrieve high quality information about mentions of functional sets of transcription factors and their target genes from PubMed abstracts. The text mining approach has not been published yet, but it has resulted in a dataset (FNL) that has been added and made available for querying in the BioGateway Beta app.

2.5.2 Why these datasets are important

BioGateway includes databases with Gene Ontology annotation data, protein data, gene data, protein-protein-interaction (PPI) data and gene transcription regulation data. By unifying them all in a common database using the same unique URIs to reference the entities, it is possible to search across several datasets to find patterns in a much more efficient manner than if one would have to manually compare the outputs from different resources.

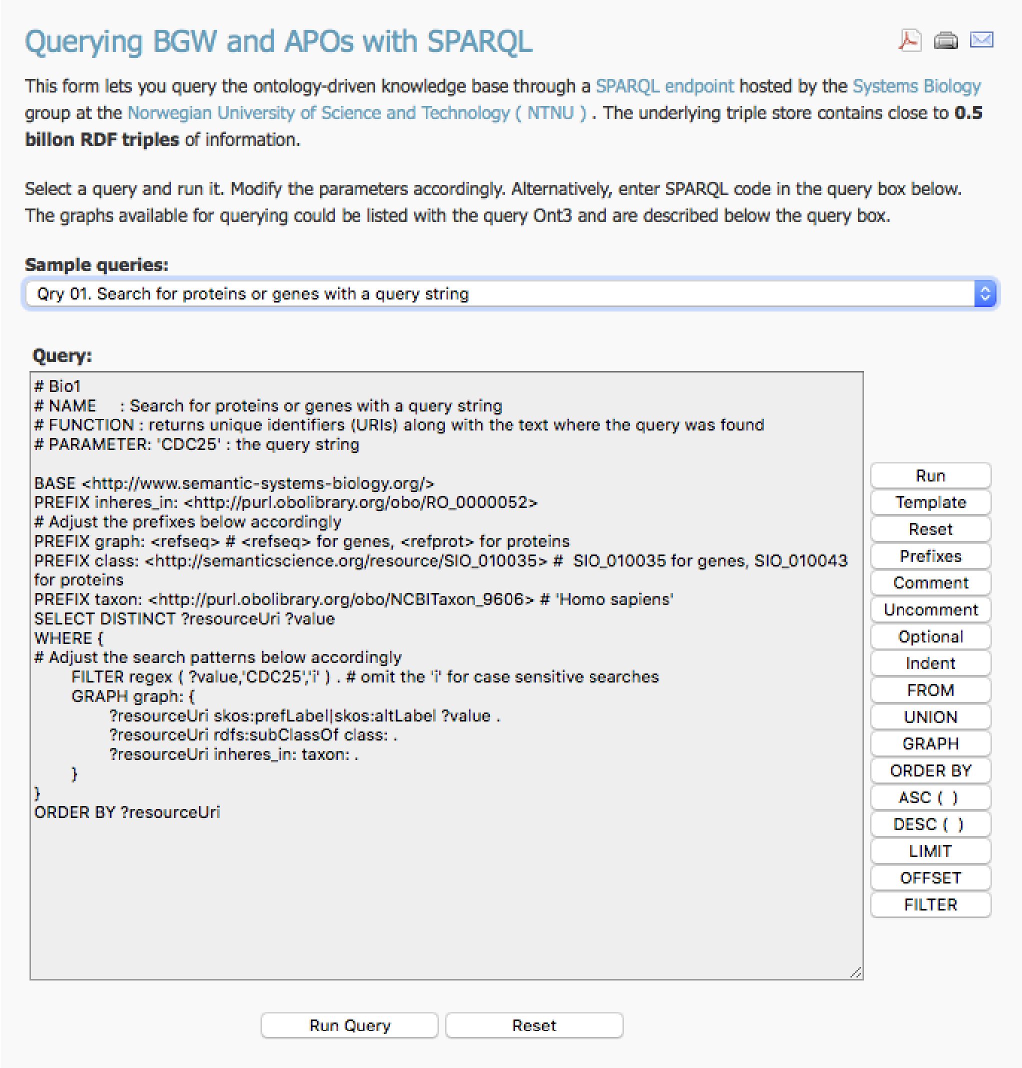
With the datasets currently available in BioGateway it is possible get the raw data of networks which describe complex biological processes of how proteins and genes interact with each other and which GO terms they are part of. Having this data available in a single resource simplifies the workflow and minimises the need to copy and paste data between various web resources.

2.5.3 SPARQL endpoint

BioGateway is currently available as a public web resource at *http://www.semantic-systems-biology.org.* It is powered by the OpenLink Virtuoso[55] SPARQL engine. It can be accessed through a SPARQL endpoint at the website[56], and data will be returned through the web browser. It also has a REST API, which enables applications to query it for data. As shown in section 1.2.2 SPARQL is a somewhat archaic query language for beginners, and might limit the target user group.

2.6 Current usage of BioGateway

Despite its potential power for supporting biological research and data interpretation, the resource is still mainly used as an example in papers about semantic web applications in Systems Biology, as a tutorial in “hackathons”, and by ontologists and bioinformaticians. The BioGateway systems administrator, Vladimir Mironov, has expressed that the current userbase seems relatively low, based on the amount of e-mails he receives when the service has been unavailable. The author himself has in fact been the first to notice server outage after several days. This would suggest



Querying BGW and APOs with SPARQL

This form lets you query the ontology-driven knowledge base through a SPARQL endpoint hosted by the Systems Biology group at the Norwegian University of Science and Technology (NTNU) . The underlying triple store contains close to **0.5 billon RDF triples** of information.

Select a query and run it. Modify the parameters accordingly. Alternatively, enter SPARQL code in the query box below. The graphs available for querying could be listed with the query Ont3 and are described below the query box.

Sample queries:

Qry 01. Search for proteins or genes with a query string

Query:

```
# Bio1
# NAME     : Search for proteins or genes with a query string
# FUNCTION : returns unique identifiers (URIs) along with the text where the query was found
# PARAMETER: 'CDC25' : the query string

BASE <http://www.semantic-systems-biology.org/>
PREFIX inheres_in: <http://purl.obolibrary.org/obo/RO_0000052>
# Adjust the prefixes below accordingly
PREFIX graph: <refseq> # <refseq> for genes, <refprot> for proteins
PREFIX class: <http://semanticscience.org/resource/SIO_010035> #  SIO_010035 for genes, SIO_010043
for proteins
PREFIX taxon: <http://purl.obolibrary.org/obo/NCBITaxon_9606> # 'Homo sapiens'
SELECT DISTINCT ?resourceUri ?value
WHERE {
# Adjust the search patterns below accordingly
      FILTER regex ( ?value,'CDC25','i' ) . # omit the 'i' for case sensitive searches
      GRAPH graph: {
            ?resourceUri skos:prefLabel|skos:altLabel ?value .
            ?resourceUri rdfs:subClassOf class: .
            ?resourceUri inheres_in: taxon: .
      }
}
ORDER BY ?resourceUri
```

Figure 2.5: The current SPARQL endpoint interface available at http://www.semantic-systems-biology.org/querying for querying BioGateway.

that BioGateway is not a common resource among the average bioinformatician, but rather a somewhat novel proof-of-concept.

Due to the somewhat steep learning curve of SPARQL[18] for those without prior experience in query languages, the current interface would be challenging for biological researchers without the time or motivation to learn SPARQL. If the current target users are in the intersection of SPARQL users and systems biologists (figure 2.6), then it does not matter if the knowledge base is relevant to the systems biologists outside of the intersection. Antezana's BioGateway paper from 2009[23] points out that *"There are various reasons for the slow pace of the adoption of Semantic Web technologies, but two of them stand out: there is a paucity of applications that demonstrate the usefulness of the Semantic Web, whereas the Semantic Web is seen as an obscure technology that is difficult to use"*, as one of the technical challenges to the BioGateway project. It also points out that visualisation tools would be one of the key objectives for further development of BioGateway. While the initial BioGateway web resource did have visualisation support for viewing results, this is no

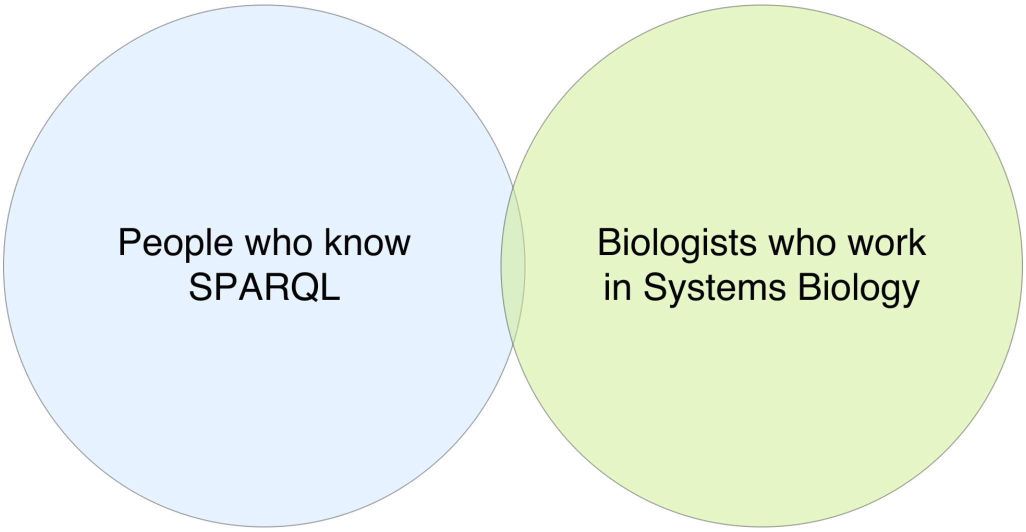


Figure 2.6: Author's illustration of the intersection of SPARQL users and biologists who work in the Systems Biology field.

longer publicly available.

We therefore proposed to find a solution for BioGateway which:

- Adds a layer that can hide the SPARQL queries from the end user and provide visualisation functionality.
- Should simplify some current workflow to provide the end user with actual value.

The focus of this thesis will therefore be to build an application which integrates visualisation functionality with an accessible user interface for performing queries, and integrate biologists and bioinformaticians in the design process to verify that the capabilities of BioGateway are valuable to them if made more accessible.

CHAPTER 3

Methodology

3.1 Use case and workflow study

To make sure that the visualisation tool is relevant for real-world applications, collaboration with potential users was essential to establish the needs and use cases where BioGateway can simplify the workflow.

Two biologists / bioinformaticians, Rafael Riudavets (RR) and Marcio Luis Acencio (MLA), associated with the Kuiper Lab[31] headed by Martin Kuiper (MK) at NTNU, have been instrumental in the design of the app, by specifying and improving workflows and choice of tools when building biological networks. They have also been testing the app during development, provided feedback and particular use cases which are representative for some of their ways of working, which provided the foundation for the design and implementation work leading to the Cytoscape app in its current state.

Their feedback is available in full in appendix 6.3.

3.1.1 Use Case 1 - RR

RR wrote his bachelor thesis in the Kuiper Lab about protein activity prediction by integrating omics data. For data gathering he mainly used resources such as KEGG[28], Reactome[57], IntAct[12], Panther[58], Signor[59] and Pathway Commons[60]. Network visualisation was done in Cytoscape.

One of the tasks in his thesis involved building a network of protein-protein interactions (PPIs) and transcription factor to target genes (TF-TG) data from various pathways known to be involved in cancer and cancer drug targeting.

RR lists the following steps in his workflow to create this network:

- Building the PPI network.
- Building the TF-TG network.
- Merging these two networks.

To build the first network of PPIs, KEGG was used to find pathways containing the proteins of interest, and the networks were then manually converted into SIF format. Then the transcription factors (TFs) in the PPI was extracted and used to find target genes (TGs) from TFactS[53] and TRRUST[54]. While these two resources support searching for TFs regulating a TG, searching the other way is not possible, so the raw datasets had to be downloaded, converted to SIF, and then filtered by the rows of the TFs of interest manually in a text editor.

The networks were then merged to produce the final network, and imported to Cytoscape to verify that no unconnected nodes were present, then exported for analysis in CausalR[61].

3.1.2 Use Case 2 - MLA

MLA is working as a biocurator at NTNU, and is currently tasked with the curating regulatory activity of TFs into structured knowledge optimised for computational analysis.

While he is currently more involved in the creation of datasets then searching through them, he has previous experience with building networks from resources. Several resources were used as well as his self-curated HTRIdb[62] dataset.

In a feedback meeting MLA also came up with the following example use case for cross-referencing GO Annotation data and TFs:

How the DbTF HNF4G is involved in tumor cell viability, colony formation and invasion?

The approach to solving this question can be summed up in the following process:

1. Having data, and have to interpret this data to generate new knowledge, using a priori knowledge.
2. Go to the biomedical literature.
    (a) Depends on just the keywords attached to the literature. Not semantically structured.
    (b) The literature is unstructured, so it's hard to find what you're looking for.
3. Using GOA acquire prior knowledge about the biological processes, molecular functions or cell parts the entities are involved in.
    (a) AmiGO or QuickGO
    (b) Looking at the GO terms the protein is associated with, trying to find terms you know to be associated with tumor cell viability, colony formation and invasion.
    (c) Found out that the protein is a known TF.
4. Look up the target genes of this TF protein.
    (a) Use a resource describing TFs, like TFactS, and target genes.
    (b) Using the target genes and their encoded proteins, return to step b. to find the GO terms these proteins are involved in.
5. Repeat step 3 and 4 until results are found or no new data is available.

This is a manual and labor-intensive process, and involves translating data from one resource to another, usually by using text files and copy-pasting the data into web interfaces.

3.2 Building a visualisation tool

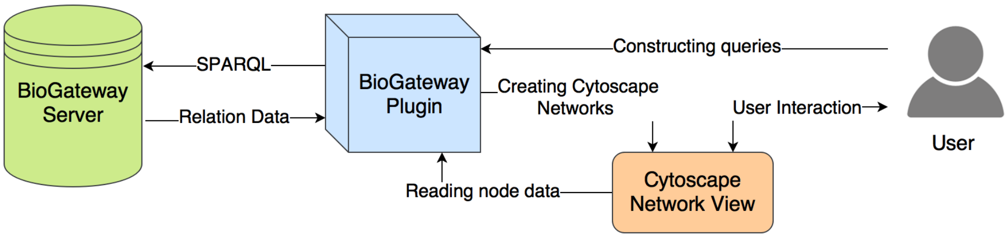


Figure 3.1: The layer of interaction between the user and the BioGateway server, hiding the underlying SPARQL and return data formats from the end user.

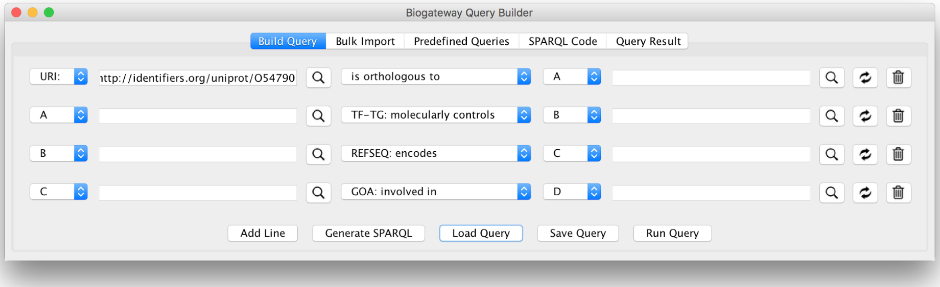
Based on the user testimonies and use cases in the previous section, the underlying functionalities required for a user-friendly interface was apparent. With the proposed solution in section 2.6 as a basis, it was decided to build a graphical user interface for interacting with BioGateway, eliminating the barrier of learning SPARQL to interact with it.

3.2.1 Deciding on the Cytoscape Platform

The choice to use Cytoscape as a visualisation platform was made early on, based on the advantages in section 2.3.1, as well as the commitment of the Kuiper Lab to contribute to this platform (BiNGO[63], CytoSQL[64]), and to teach the use of this platform in the Systems Biology course *"Systems Biology: Resources, standards and tools"*, BI3019[65].

By enabling users access to BioGateway from within a network building tool, it was also possible to automatically import and update the data being queried directly into the networks being worked with, removing this step from the workflow as illustrated in figure 3.1.

3.2.2 Query Builder Tool



URI: ıttp://identifiers.org/uniprot/O54790 is orthologous to A
A TF-TG: molecularly controls B
B REFSEQ: encodes C
C GOA: involved in D

Figure 3.2: The Query Builder tool allows users to construct SPARQL-like queries without knowing SPARQL.

The BioGateway Query Builder, shown in figure 3.2, was designed to allow users to construct more complex queries than searching for a single type of relation. By combining several lines of relations in a format similar to relationships in the RDF format described in section 1.2.1, with variables that represent the values being searched for, complex relationships can be formulated to search for a specific intersection of relationships.

3.2.3 XML Config File

As mentioned in section 2.4, similar attempts at making RDF plugins for Cytoscape has been made in the past, but a lack of updates has made them less relevant today. In order to future-proof the BioGateway app and make it less dependent on maintenance by the original developer or others with an understanding of the inner workings of the application code, an XML configuration file is used to configure the application when it is loaded. Future maintainers of the app would only need to understand the XML file in order to reconfigure it, and would not need to set up the full Cytoscape app development environment needed to compile a new version.

This XML file is hosted by the maintainers of BioGateway, and allows some of the client functionality to be updated to match new features on the server. This approach also reduces the need for users to update the app itself to change minor features.

3.2.4 Predefined Queries

Based on the idea of example queries in the BioGateway SPARQL web interface, a set of specific queries are provided for various types of searches. These queries are defined in the configuration XML file, and the app will dynamically generate a GUI based on the parameters defined for the queries. These queries are subject to change in near future, so examples will not be given here.

3.2.5 Contextual right-click menus

For easy and efficient manipulation of nodes and networks, some queries was enabled by right-clicking a node in a network view, giving the user the ability to search for relations to or from the selected node, automatically getting the results back into the network. This was important to let users "browse" through BioGateway for relevant nodes and relations, and more quickly build the network of interest. Similar right-click query functionality has also been present in the other RDF-based Cytoscape plugins described in section 2.4, and it would be advantageous to include user-interface elements that users could be familiar with and expect to be present.

3.3 App development and beta testing

The Cytoscape plugin was developed in close collaboration with the bioinformatics community at NTNU, and feedback was given on a regular basis. RR and MLA were beta testers of the app, and provided invaluable feedback from their systems biologists point of view, as the author himself is from a computer science background.

3.3.1 Usage in BI3019 (NTNU)

The application was demonstrated for students in the BI3019 course taught by MK. It was introduced as a prototype that could be used alongside other conventional tools such as web resources and the BiNGO Cytoscape app[63], to build a network of interactions between a predefined set of genes.

The need to work with large sets of genes at the same time highlighted a very obvious shortcoming of the app at the time; it was only able to search for a single entity at the time. Within a few days the Bulk Importer (see section 5.3) was introduced, allowing users to quickly import a set of genes into a BioGateway-compatible Cytoscape network. Working with larger (50 - 200) sets of nodes also gave inspiration to creating the “Multi-node queries” (see 5.4.5), enabling users to perform the same query on all of the selected nodes at once. Prior to this, the focus had been more on the query builder, but this new set of features enabled new, more efficient ways of using the current network to build queries, instead of just the query builder.

3.3.2 Beta user input

RR and MLA were both clear that they felt that the way BioGateway represents Protein-Protein Interactions (PPIs) was a bit tedious, as it models each interaction as its own node, with the proteins involved annotated as the agents of the PPI. This makes a lot of sense when multiple proteins are involved in the same PPI, but makes networks of binary PPIs more cluttered. A client-side workaround was implemented, where the app will enable users to query for only binary PPIs, and converts the result into a binary PPI relation between the two proteins, removing the PPI node separating them. This made binary PPI searches in BioGateway more viable, expanding the usability to use cases involving finding PPIs.

CHAPTER 4

Results and Evaluation

Using the two examples given by Rafael Riudavets (RR) and Marcio Luis Acencio (MLA) in 3.1, the BioGateway app can be used to look for relevant results. The functionality used here is explained in more detail in the App Manual in chapter 5.

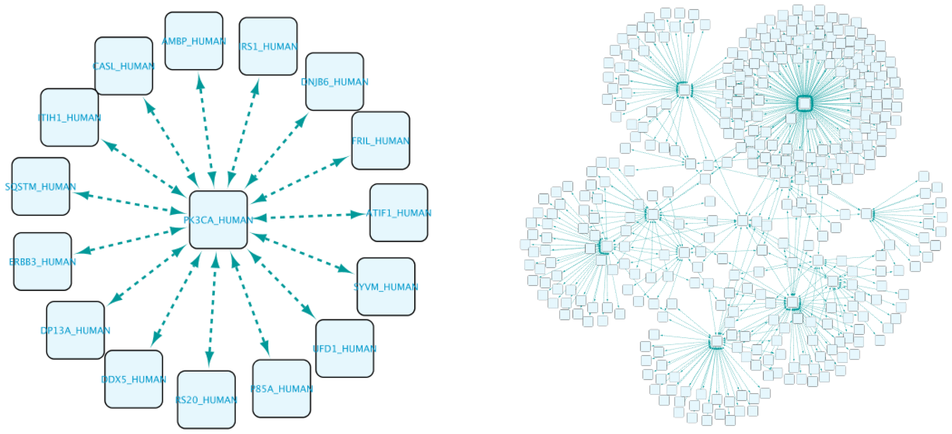
4.1 Use Cases with the BioGateway App

4.1.1 Creating a PPI network and finding TF-TG relations

Looking for the effects of a drug inhibiting the function of the PIK3CA_HUMAN protein, using transcriptomics.

Starting by adding the *PIK3CA_HUMAN* protein to a new network. As we are interested in transcription factors, we search for PPIs between this protein and transcription factors. A search for all binary PPIs involving *PIK3CA_HUMAN* gives us the network in figure 4.1a.

Figure 4.1: Protein-protein interaction network



(a) A network with the *PIK3CA_HUMAN* protein and PPIs to proteins it interacts with.

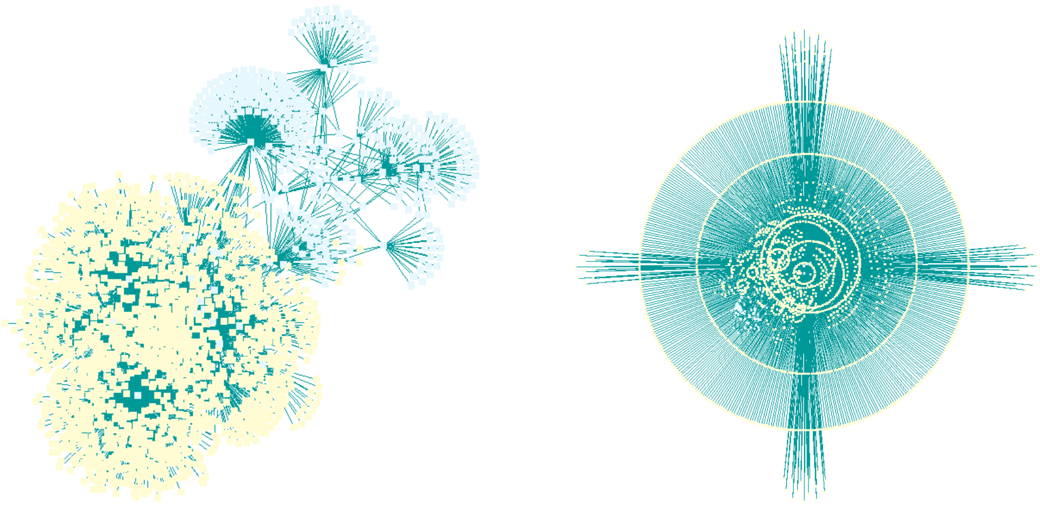
(b) Second set of PPIs.

Next, we check if any of these new proteins are TFs by selecting all nodes, right-clicking and choosing *"Fetch relations FROM selected"* → *"molecularly controls"*.

This does not return any results, so it seems none of the new proteins are TFs. Selecting all the nodes and searching for PPIs (shown in 5.4.5) from all of them again, gives a larger network shown in figure 4.1b.

Repeating the process by selecting all the nodes and searching for TFs again returns 1763 TFTG relations and importing all of them gives the network in figure 4.2a.

Figure 4.2: Protein-protein interaction network with target genes



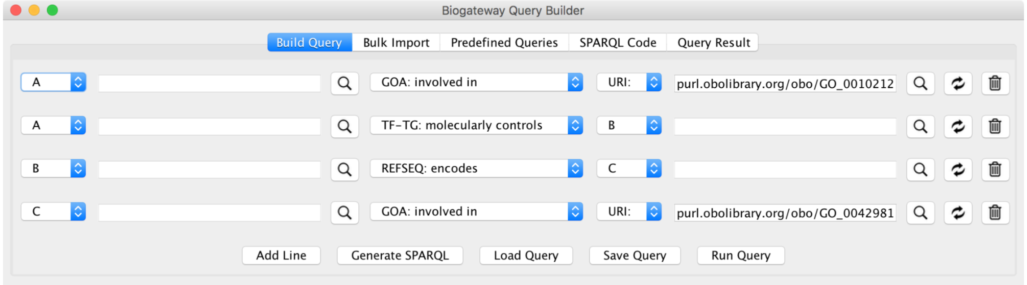
(a) With target genes from the TFs in 4.1b.

(b) Final network after removing redundant leaf proteins.

We use Cytoscape's built-in filtering feature to remove all proteins that are connected to only one other node, as these are no longer of interest. The resulting network (figure 4.2b) contains the TFs and genes of interest, and can be exported to SIF format for further statistical analysis in CausalR.

4.1.2 Finding TFs annotated by a set of GO terms

Create a network of TFs involved in "response to ionizing radiation" that are regulating genes encoding proteins involved in "regulation of apoptotic process".



A		GOA: involved in	URI:	purl.obolibrary.org/obo/GO_0010212
A		TF-TG: molecularly controls	B	
B		REFSEQ: encodes	C	
C		GOA: involved in	URI:	purl.obolibrary.org/obo/GO_0042981

Figure 4.3: The query for finding TFs involved in the terms *"response to ionizing radiation"* and *"regulation of apoptotic process"*.

We can formulate these relationships as shown in figure 4.4. This can be formulated as a query in the Query Builder, as shown in figure 4.3. First, the URIs for the GO term *"response to ionizing radiation"* is found using the Node Lookup (described in 5.2.2). The variable *A* is representing the nodes which are involved in this term, and we specify that *A* should also *molecularly control* nodes *B*. As the TF-TG datasets contain relations between TFs and genes, *B* would be genes, and we will expand the query with the nodes *C encoded* by *B*. Finally, the URI for the GO term *"regulation of apoptotic process"* is found with the Node Lookup tool, and the valid values of *C* will be constrained to the nodes that are *involved in* this specific GO term. After running the query we import all the relations into a new network.

To check if any of the proteins are involved in both the GO terms, we select all the relations, right-click and select *"Find common relations FROM selected"* → *"GOA: involved in"*, and set the minimum common relations to 2. This will return a list of GO terms that at least 2 of the proteins in the set have in common. By pressing *"Import relations between existing nodes"*, we discover one new relation; *BRCA1_HUMAN* is involved in both of the terms of interest (figure 4.5a).

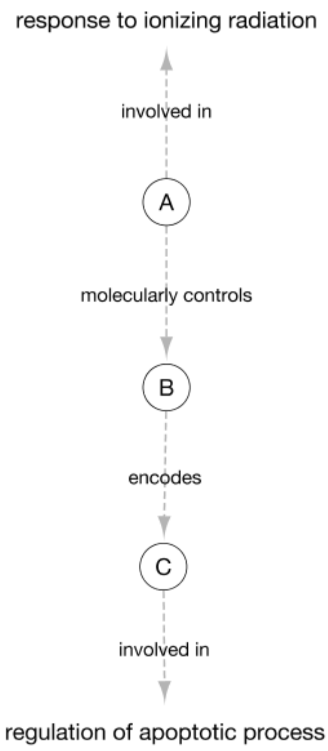
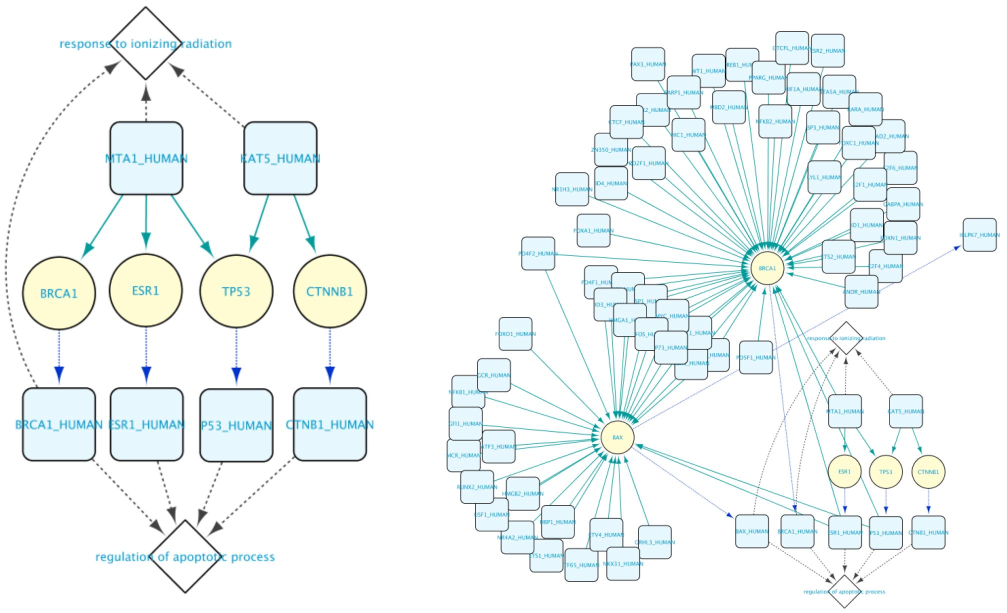

response to ionizing radiation
involved in
A
molecularly controls
B
encodes
C
involved in
regulation of apoptotic process

Figure 4.4: Illustration of the relations between the variables in figure 4.3

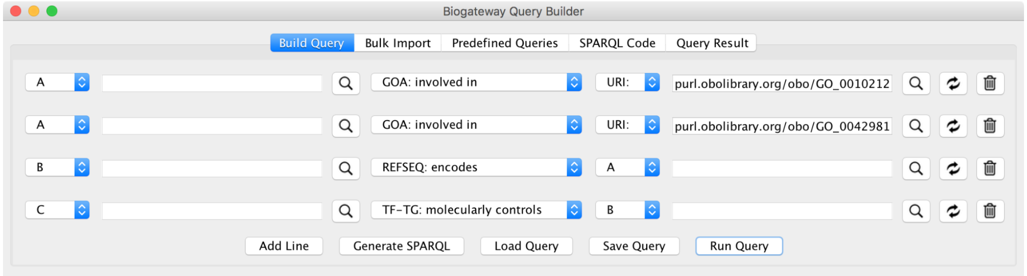
Figure 4.5: Interaction networks with the TFs regulating genes involved in both GO terms



(a) First network including the results from the first query.

(b) Network after the results in the second query has been added.

If we want to expand the network to involve all TFs that regulating genes that encode for proteins involved in both of the terms, we can formulate a new query as shown in figure 4.6.



A	GOA: involved in	URI:	purl.obolibrary.org/obo/GO_0010212
A	GOA: involved in	URI:	purl.obolibrary.org/obo/GO_0042981
B	REFSEQ: encodes	A	
C	TF–TG: molecularly controls	B	

Figure 4.6: Searching for the TFs regulating genes encoding proteins involved in both GO terms of interest.

After running the query and importing the results into the current network, we can see that a new gene, *BAS*, was found. We have now included all:

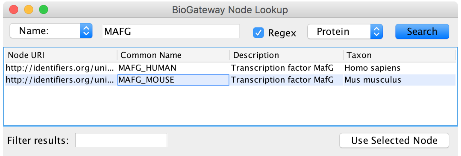
- TFs annotated as involved in *"response to ionizing radiation"* and are regulating genes which encode for proteins involved in *"regulation of apoptotic process"*, and
- The TFs that regulate genes encoding proteins that are involved in both terms.

This network can now be used for further analysis and querying.

4.1.3 Connection between TF and oxidative stress

Why is the expression of DbTF gene MafG induced by cigarette smoke condensate in mice?

This use case was proposed by MLA in section 3.1.2. Because we only have GOA data for proteins, we are initially interested in the *MAFG_MOUSE* protein, and start building a query with it by adding it with the Node Lookup feature (figure 4.7).



Name: MAFG ☑ Regex Protein

Node URI	Common Name	Description	Taxon
http://identifiers.org/uni...	MAFG_HUMAN	Transcription factor MafG	Homo sapiens
http://identifiers.org/uni...	MAFG_MOUSE	Transcription factor MafG	Mus musculus

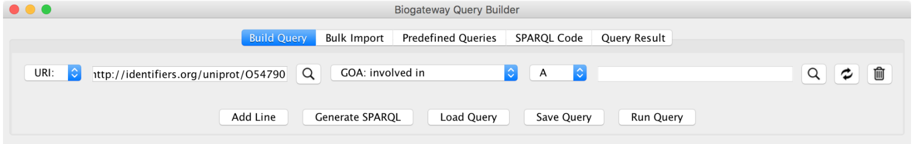
Filter results:

Figure 4.7: Looking up the URI for *MAFG_MOUSE*.

As we are looking for Gene Ontology Annotations, the "GOA: involved in" relation is selected, and the query is executed (figure 4.8).

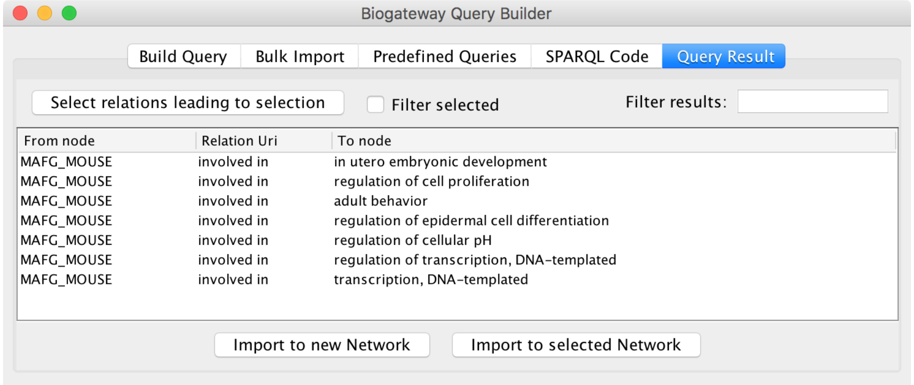
As shown in figure 4.9, his did not provide any data related to oxidative stress, associated with cigarette smoke. Attempting to look for TF-TG relations for the same protein did not provide any results, although it is known to be a TF. Searching in other species might yield more results.

The next query (figure 4.10) will first look for any protein A, which is orthologous to (equivalent to in another organism) the *MAFG_MOUSE* protein. Then, if any of these proteins A are a TF, find all target genes B, and the proteins they encode for, C. Then look for any GO term that the proteins C are part of.



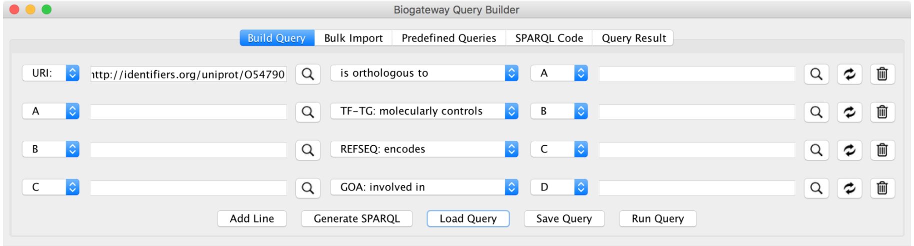
URI: ıttp://identifiers.org/uniprot/O54790 | GOA: involved in | A

Figure 4.8: Looking for terms that *MAFG_MOUSE* is involved in.



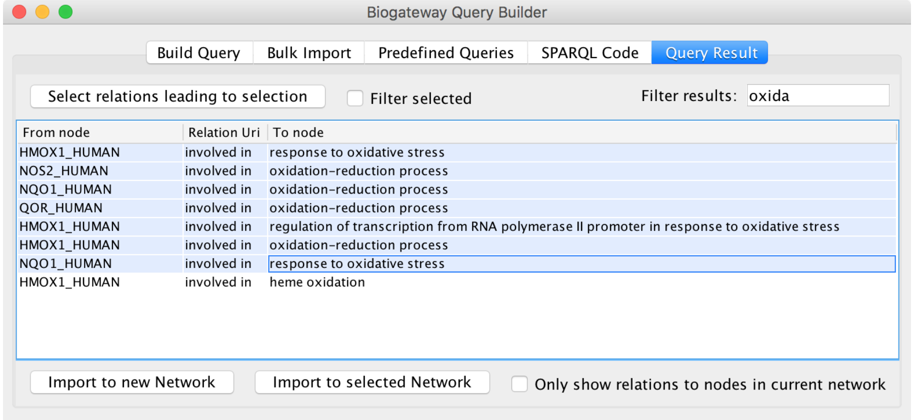
From node	Relation Uri	To node
MAFG_MOUSE	involved in	in utero embryonic development
MAFG_MOUSE	involved in	regulation of cell proliferation
MAFG_MOUSE	involved in	adult behavior
MAFG_MOUSE	involved in	regulation of epidermal cell differentiation
MAFG_MOUSE	involved in	regulation of cellular pH
MAFG_MOUSE	involved in	regulation of transcription, DNA-templated
MAFG_MOUSE	involved in	transcription, DNA-templated

Figure 4.9: Annotation data on the MAFG protein in mice is somewhat limited.



URI: ıttp://identifiers.org/uniprot/O54790 | is orthologous to | A
A | TF-TG: molecularly controls | B
B | REFSEQ: encodes | C
C | GOA: involved in | D

Figure 4.10: A second query, looking for TFs in orthologous proteins, and the terms their targets are involved in.



Filter results: oxida

From node	Relation Uri	To node
HMOX1_HUMAN	involved in	response to oxidative stress
NOS2_HUMAN	involved in	oxidation-reduction process
NQO1_HUMAN	involved in	oxidation-reduction process
QOR_HUMAN	involved in	oxidation-reduction process
HMOX1_HUMAN	involved in	regulation of transcription from RNA polymerase II promoter in response to oxidative stress
HMOX1_HUMAN	involved in	oxidation-reduction process
NQO1_HUMAN	involved in	response to oxidative stress
HMOX1_HUMAN	involved in	heme oxidation

Figure 4.11: Filtered results from the query in 4.10.

This results in a much larger set of terms available for the human version of the protein. Filtering by terms starting with “oxida”, we find a set of interesting relations (figure 4.11). Then, by clicking the “Select relations leading to selected” button in the upper right corner, all relations leading to this set is also selected, so they can be included, and imported to a new network. In figure 4.12 we see a network of 4 proteins of interest for finding a correlation between the MAFG protein and oxidative stress.

Repeating the process with PPI interactions instead of TF-TG (figure 4.13) reveals an additional protein and the PPIs linking it with *MAFG_HUMAN* (figure 4.14). Note that the workaround to search for binary PPIs directly is not available in the query builder, so it is necessary to search for the PPIs containing them. Also, the *"Select relations leading to selected"* feature will not select the relations from the PPIs to the *MAFG_HUMAN* protein representing the variable A, as the relation goes from B to A, so these relations must be selected manually when using PPIs.

This entire process can be completed in minutes, and does not involve any copying or parsing of results between web resources. Neither does it require the user to understand advanced query languages such as SPARQL. The resulting network might not be a complete representation of all knowledge available, as BioGateway does not contain data from all common datasets, and further querying into other databases might be required.

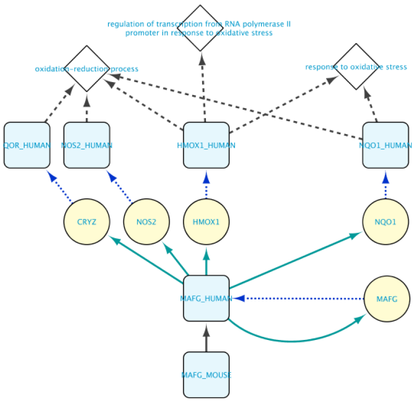
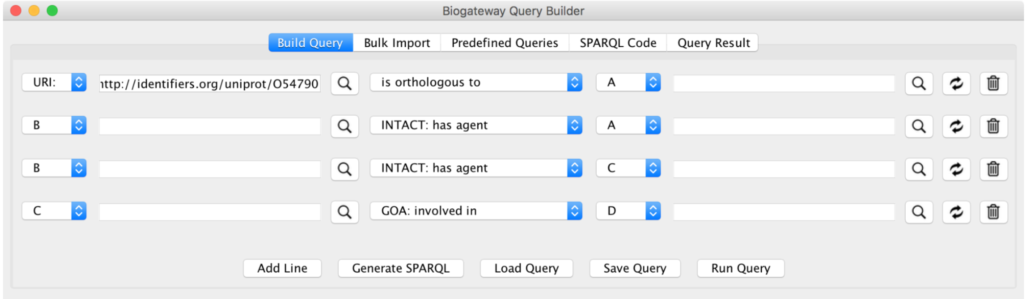

regulation of transcription from RNA polymerase II promoter in response to oxidative stress
oxidation-reduction process
response to oxidative stress
QOR_HUMAN
NOS2_HUMAN
HMOX1_HUMAN
NQO1_HUMAN
CRYZ
NOS2
HMOX1
NQO1
MAFG_HUMAN
MAFG
MAFG_MOUSE

Figure 4.12: The resulting network after importing the results from the query in 4.11


Biogateway Query Builder

Build Query | Bulk Import | Predefined Queries | SPARQL Code | Query Result

URI: ıttp://identifiers.org/uniprot/O54790 | is orthologous to | A
B | INTACT: has agent | A
B | INTACT: has agent | C
C | GOA: involved in | D

Figure 4.13: Querying for proteins that are involved in the same PPIs as proteins orthologous to *MAFG_MOUSE*, and the GO terms they are involved in.

4.2 Evaluation

4.2.1 First impressions

The first impressions from the test users have been very positive. They have highlighted that expanding the network as-you-go inside Cytoscape is much easier than having to search through web resources for datasets to download and then import into Cytoscape manually.

While Cytoscape allows you to download networks from popular resources, you would have to download the whole predefined networks and afterwards filter it, instead of building it out from the nodes of interest. This often results in first downloading huge networks, and then having to filter out the needles in the haystack.

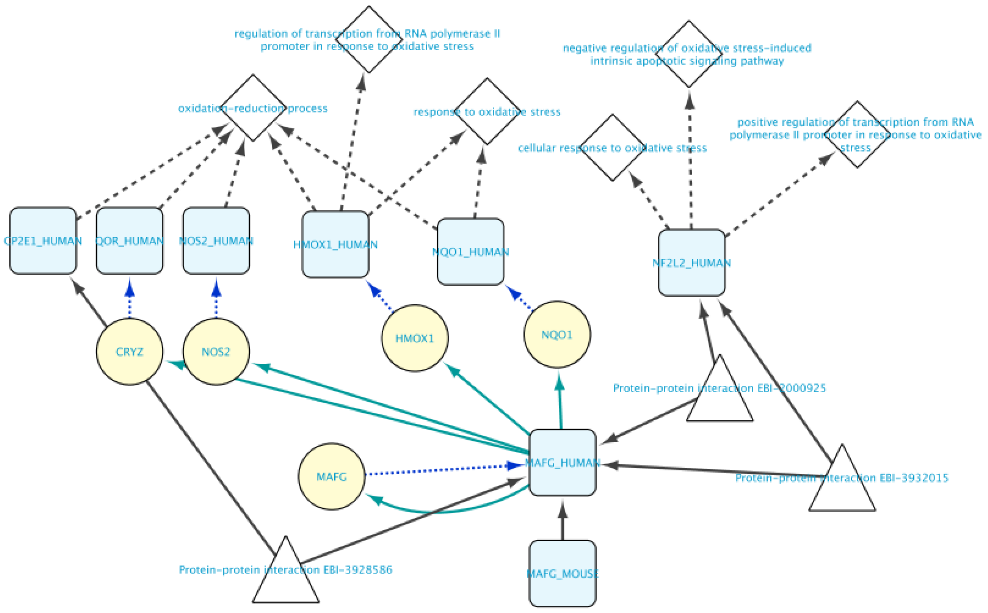

regulation of transcription from RNA polymerase II promoter in response to oxidative stress
negative regulation of oxidative stress-induced intrinsic apoptotic signaling pathway
oxidation-reduction process
response to oxidative stress
cellular response to oxidative stress
positive regulation of transcription from RNA polymerase II promoter in response to oxidative stress
CP2E1_HUMAN
QOR_HUMAN
NOS2_HUMAN
HMOX1_HUMAN
NQO1_HUMAN
NF2L2_HUMAN
CRYZ
NOS2
HMOX1
NQO1
Protein-protein interaction EBI-2000925
MAFG
MAFG_HUMAN
Protein-protein interaction EBI-3932015
Protein-protein interaction EBI-3928586
MAFG_MOUSE

Figure 4.14: The final network with both TF-TG and PPI data.

The ability to perform these queries on multiple nodes at once is also a popular feature, and the ability to find relations that the selected nodes have in common is used as a feature to quickly find new relationships between the existing nodes in the network, or find the entities that connect them together.

The client-side workaround to infer binary protein-protein interactions (PPI) is also a welcome feature, and is essential in use cases where looking for PPIs is one of the steps.

4.2.2 User Interface Challenges

MLA pointed out several elements of the user interface (UI) that he feels has room for improvement.

The first issue is regarding how to search for genes and proteins. Many bioinformaticians interchangeably uses the same names for genes and the proteins they encode for. In his example, when searching for the protein encoded by the well-known TP53 gene, he would expect to find something. However, as BioGateway has labeled this protein *TP53_HUMAN*, a direct match is not immediately found. While it would certainly be possible to make the app hide this distinction, a choice has to be made between:

- Simplifying the application to make it easier to use, but risk oversimplifying it too much by hiding important distinctions in the edge cases where the gene and protein associations are not one-to-one, or,

- Keep the distinction between proteins and genes, but force users to go to the extra step of including both genes and proteins in the queries they build and add extra clutter to the networks produced.

So far, the application is leaning towards the latter, and will sacrifice usability for accuracy in this case.

MLA's second UI related concern is with how search results are shown when searching for nodes to add to the Query Builder. The search results includes the URI, the common label matching the search term, and a description. However, the label and description is not always enough for the user to make sense of which entity they are looking for, especially if the labels are not the same as they are used to. The labels used in the descriptions does not always match the labels used for the nodes themselves, adding up to the confusion. This is an artifact of the data graphs in BioGateway and the data sources used to build these graphs, and would be challenging to address.

The final concern is regarding PPIs and how the BioGateway server stores them - or rather does not store binary PPI interactions as relations. The app does have "helper functions" to infer binary PPIs to a *"molecularly interacts with"* relation on the fly, but this does not work in the Query Builder. If the *"molecularly interacts with"* relations - or a similar relation type - were inferred and included by the BioGateway server, this problem would be resolved.

4.2.3 Support for networks created outside the app

Users who are building networks with the help of other Cytoscape apps, or are importing the networks, have been disappointed to learn that these networks are not compatible with the BioGateway app. Because BioGateway is using node URIs as a foundation to identify the entities being reasoned on, it is absolutely dependant on having these node URIs present in the Cytoscape networks that the app will interact with. Unfortunately, other Cytoscape apps which create and modify networks normally does not include this field, making the BioGateway app incompatible with networks created by other apps. If the developers of other apps would decide to use node URIs as one of the fields in their network, this could be resolved, but this would be a lot to ask for unless there is a wider push towards Semantic Web practices such as URIs in the bioinformatics community.

It might be more realistic to propose adding an "import network" feature to the app, helping users to look up the URIs associated with the entities in the network and add them as properties to the nodes and edges. However, such a tool would have to be constructed to deal with a lot of ambiguities such as when several entities share a name, when proteins are annotated by their gene name, when the relation types are not specific enough, and so forth. Such a tool could require significant time and effort to implement.

4.2.4 BioGateway Database feedback

The BioGateway server and app is not of much value to the user if the data they need is not in available. A common response from potential users who are intrigued by the performance of the BioGateway app has been *"it sounds great, but I need support for this specific dataset"*.

MLA has been clear about this in his feedback, listing several datasets he would like to see included in the future, mentioned in section 6.2.

There has also been some concern about the lack of experimental source data for the relations in some of the dataset graphs in BioGateway. This would be important to address in order to let the users verify the experimental foundation and credibility of the relations in their networks.

4.2.5 Pitfalls of oversimplifications

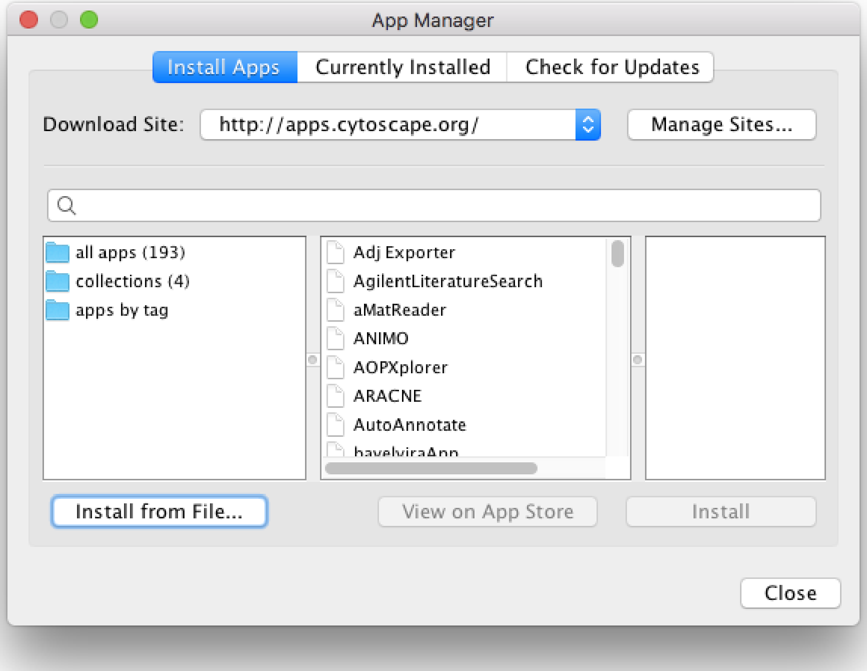
While making the BioGateway app easy to use would make it accessible for a broader audience, some concern should be given to what happens if the tool becomes an alternative to using the dataset publishers' websites altogether. If used by users who do not know how the different datasets are structured, whether they come from trusted, human-annotated sources or computationally mined from texts, and how reliable the sources are, they might end up believing the false positives or negatives provided by BioGateway. This may be remedied by giving users access to the full provenance with all the source data (PubMed IDs, database sources, dates of database access), and by providing particular quality criteria that can be used for selecting and filtering specific data elements, as well as visualising confidence.

CHAPTER 5

App Manual

5.1 Installation

Download the BioGateway plugin .jar file, and save it to disk.



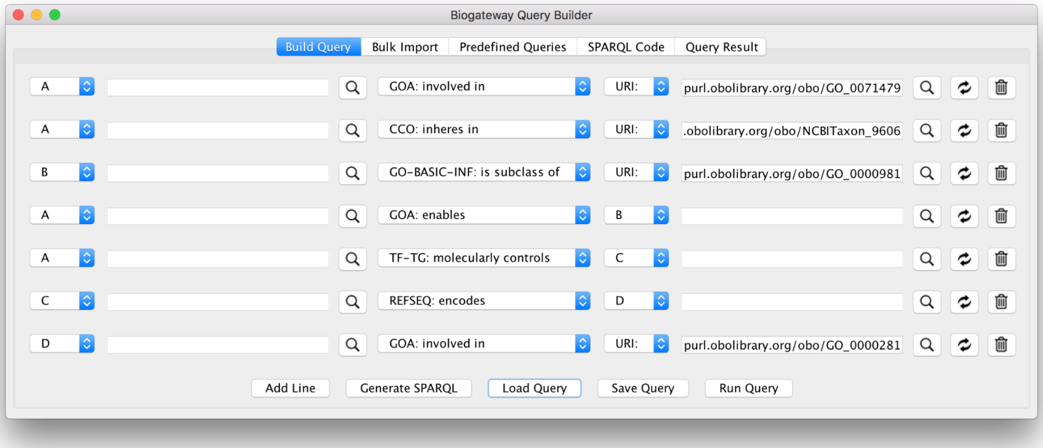
Download Site: http://apps.cytoscape.org/

Figure 5.1: Installing through the Cytoscape App Manager

Open the App Manager by choosing *"App Manager..."* under the *"Apps"* menu in Cytoscape. Press *"Install from File. . . "* and locate the .jar file that was downloaded.

5.2 The Query Builder

The Query Builder is a powerful search tool that lets the user construct complex queries with several variables to search for. By combining specific biological entities,



A		GOA: involved in	URI:	purl.obolibrary.org/obo/GO_0071479
A		CCO: inheres in	URI:	.obolibrary.org/obo/NCBITaxon_9606
B		GO-BASIC-INF: is subclass of	URI:	purl.obolibrary.org/obo/GO_0000981
A		GOA: enables	B	
A		TF-TG: molecularly controls	C	
C		REFSEQ: encodes	D	
D		GOA: involved in	URI:	purl.obolibrary.org/obo/GO_0000281

Figure 5.2: The BioGateway Query Builder.

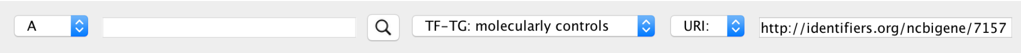
which we will name *"nodes"*, or by specifying variables, the user is able to find results that depend on several factors at once. To open the Query Builder, click the *"Apps"* menu, and select *"BioGateway"* → *"Create query"*.

5.2.1 Constructing Queries

A query consists of one or more rows, each row representing a relation between two entities. The nodes can be defined in one of two ways; as a URI representing an actual entity in the BioGateway database, such as a protein or a gene, or as a variable represented by a letter such as A, B, C, etc. The pulldown menus to the left of the text fields lets the user pick between *"URI:"* or assign the node to a variable letter.

The ordering of the nodes are important; each relation goes from the node left of it, to the node that is on the right side. The the 4th line of figure 5.2 specifies that *A enables B*, not the other way around.

To get additional lines, click the *"Add Line"* button in the lower left corner. To remove a line, click the garbage can icon on the far right side. To swap the node parameters of the left and right side of the relation to change the directionality of the relation, click the swap icon next to the garbage icon.



A		TF-TG: molecularly controls	URI:	http://identifiers.org/ncbigene/7157

Figure 5.3: A query line.

Node URIs

When a node is set to be represented by an URI, the value in the text field will be used. All URIs starts with *"http://"*, and are valid web locations which describe the biological entity represented. The URI must be in the BioGateway database to work correctly. In order to get the correct URI for an entity, the Node Lookup view, described in section 5.2.2, can be used by clicking the magnifying glass next to the

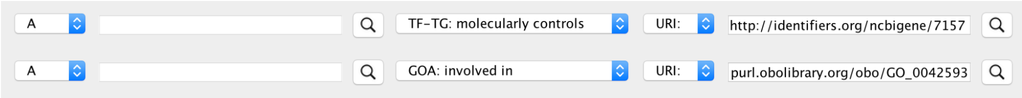
URI text field. The URIs are also available in *"identifier uri"* column of node data tables created by the BioGateway plugin.

To see the name of the entity represented by an URI, hover the mouse over the URI field. If the entity has been loaded from the server, it will be shown in the mouseover tooltip.

Variable letters

To denote any matching node, variable letters should be used. The variable letter *"A"* would mean *"any node in the database which satisfies the conditions..."*, where the conditions are the relations in the rest of the query. In figure 5.3, the variable *"A"* would represent all nodes which have the relation *"molecularly controls"* pointed to the node with the URI *"http://identifiers.org/ncbigene/7157"*, in this case the URI for the *TP53* gene.

Combining multiple lines



A | | TF-TG: molecularly controls | URI: | http://identifiers.org/ncbigene/7157

A | | GOA: involved in | URI: | purl.obolibrary.org/obo/GO_0042593

Figure 5.4: Variables over multiple lines.

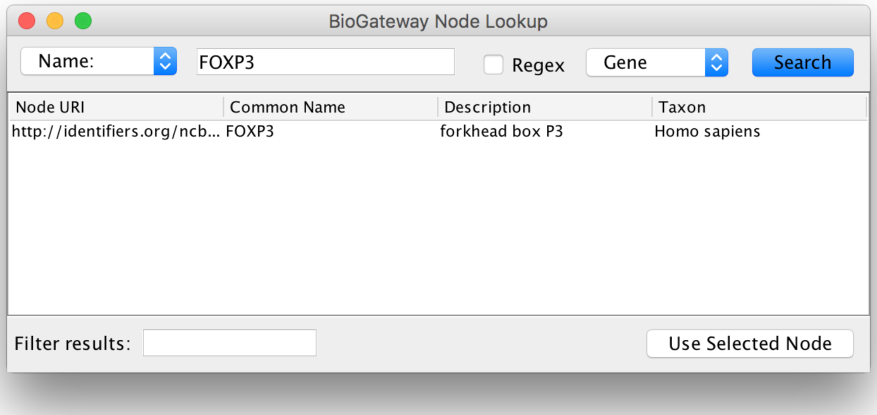
While getting all the nodes with a specific relation to a specific node is useful, the real functionality in the Query Builder is the ability to combine several lines to get more specific results. By involving the same variable in several relations, only the nodes which satisfy all conditions are returned. In figure 5.4, the query from figure 5.3 has been expanded with a new relation, constraining the results to the entities which also are involved in a GO term, in this case *"glucose homeostasis"*, so the results would be all transcription factors for the *TP53* gene which are also involved in *"glucose homeostasis"*.

Queries can span several lines, and have variables on both sides of the relations. For instance, the query in figure 5.2 looks for all nodes A which:

1. Are involved in the *GO:0071479* term.
2. Inheres in *"homo sapiens"*.
3. Enables the GO term B.
4. Molecularly controls the gene C.

Where:

1. B is a subclass of the *GO:0000981* term.
2. C encodes the protein D.
3. D is involved in the *GO:0000281* term.



BioGateway Node Lookup

Name: | FOXP3 | □ Regex | Gene | Search

Node URI	Common Name	Description	Taxon
http://identifiers.org/ncb...	FOXP3	forkhead box P3	Homo sapiens

Filter results: | Use Selected Node

Figure 5.5: Looking up the FOXP3 gene.

5.2.2 Using the Node Lookup View

To easily look up a node URI, click the magnifying glass in figure 5.4 next to the URI text field. The Node Lookup view has several ways of searching for a node, which can be selected in the drop-down menu in the upper left corner. To use one of the resulting nodes, select it and click “Use Selected Node”. If many results are found, the filter text field in the lower left corner can be used to limit the results.

Search by Name

In this mode, the BioGateway will search for nodes matching the name in the text field.

Regex The *“Regex”* checkbox indicates if BioGateway should search for partial matches to the search text. The regex search will match with anything that contains the search text. Normal regex symbols such as are usable here. The most notable symbols are . and *, meaning *“any character”* and *“zero or more occurrences of the previous character”*. Combining them to *“.*”* means *“zero or more of any character”*, which can be useful when searching. For example, searching for GO terms named *“response radiation”* will not match anything, while *“response.*radiation”* will return all terms with *“response”* and *“radiation”* in their name.

Regex searches are much more resource-intensive than exact matches, and takes much longer time. Searches that are too broad might have too many results, and fail to complete.

Entity Type The type of biological entity to search for must be specified. The available types are:

- Protein
- Gene

- GO Term
- Taxon

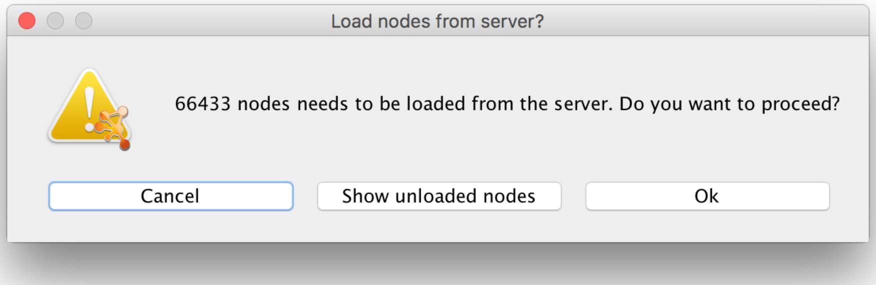
Note that when searching for taxon, searches are with regex by default.

Search by URI This allows the user to verify the existence and name of a URI, and will try to fetch the node with this specific URI. There will only be one result. This mode is also useful when using the Node Lookup view from within a Cytoscape network to add new nodes.

Search by UniProt ID A URI will be generated by appending the UniProt ID to *"http://identifiers.org/uniprot/"*. For example, when looking up the UniProt ID *"P04637"*, it will generate the URI *"http://identifiers.org/uniprot/P04637"*. The URI lookup is the same as the URI search above.

Search by GO Term The term must be on the form *"GO: ... "*. A URI will be generated from the GO term. As an example, the term *"GO:0003674"* will result in a search for the URI *"http://purl.obolibrary.org/obo/GO_0003674"*. The URI lookup is the same as above.

5.2.3 Running queries



Load nodes from server?

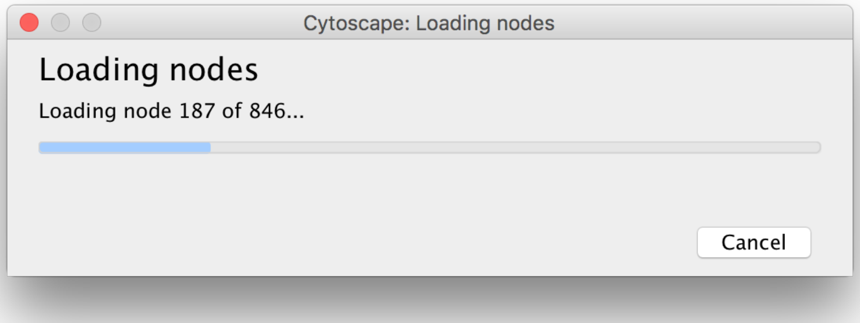
66433 nodes needs to be loaded from the server. Do you want to proceed?

Figure 5.6: A warning about a large result set that will take time to load.

After clicking the *"Run Query"* button in the Query Builder, the query will be generated and sent to BioGateway. When executing a query, the results are loaded in several steps. If a result set is particularly huge, it might return thousands of nodes. The BioGateway app will try to reuse previously loaded nodes to avoid additional loading times, but will warn the user if more than 1000 nodes needs to be loaded, as shown in figure 5.6.

The warning will let the user cancel the query, proceed with loading all the nodes, or just show the nodes as URIs without loading their names and descriptions. This can be useful when only the nodes already loaded are needed.

When the results are fully loaded, or if the user decides to not load all nodes, the Query Result window appears.

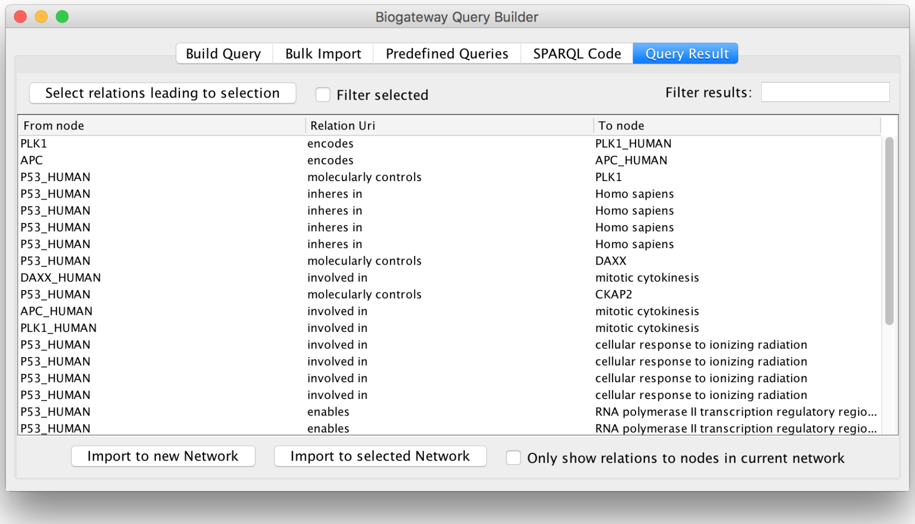


Loading nodes

Loading node 187 of 846...

Figure 5.7: In progress of loading nodes from the server.

5.2.4 Query Results



□ Filter selected

Filter results:

From node	Relation Uri	To node
PLK1	encodes	PLK1_HUMAN
APC	encodes	APC_HUMAN
P53_HUMAN	molecularly controls	PLK1
P53_HUMAN	inheres in	Homo sapiens
P53_HUMAN	inheres in	Homo sapiens
P53_HUMAN	inheres in	Homo sapiens
P53_HUMAN	inheres in	Homo sapiens
P53_HUMAN	molecularly controls	DAXX
DAXX_HUMAN	involved in	mitotic cytokinesis
P53_HUMAN	molecularly controls	CKAP2
APC_HUMAN	involved in	mitotic cytokinesis
PLK1_HUMAN	involved in	mitotic cytokinesis
P53_HUMAN	involved in	cellular response to ionizing radiation
P53_HUMAN	involved in	cellular response to ionizing radiation
P53_HUMAN	involved in	cellular response to ionizing radiation
P53_HUMAN	involved in	cellular response to ionizing radiation
P53_HUMAN	enables	RNA polymerase II transcription regulatory regio...
P53_HUMAN	enables	RNA polymerase II transcription regulatory regio...

□ Only show relations to nodes in current network

Figure 5.8: The results after running the query in figure 5.2.

The results from the query will be a set of all relevant relationships found between the matching nodes. If the query was on the form of $A \to B \to C \to D$, the relations between $A \to B$, $B \to C$, and $C \to D$ will be shown individually.

In the cases where a large amount of results are found, the *"Filter results"* text field can be used to limit the result set.

Select upstream relations

This button will select all relations that are pointing to the *"From node"* in one of the selected relations, recursively. This feature is useful for finding the relevant path of relations leading to the relations selected, so they can be included in the network.

Filter selected

This checkbox toggles the filtering to only show the selected nodes, instead of filtering by text search. This can be useful to see the set of relations that are about to be imported.

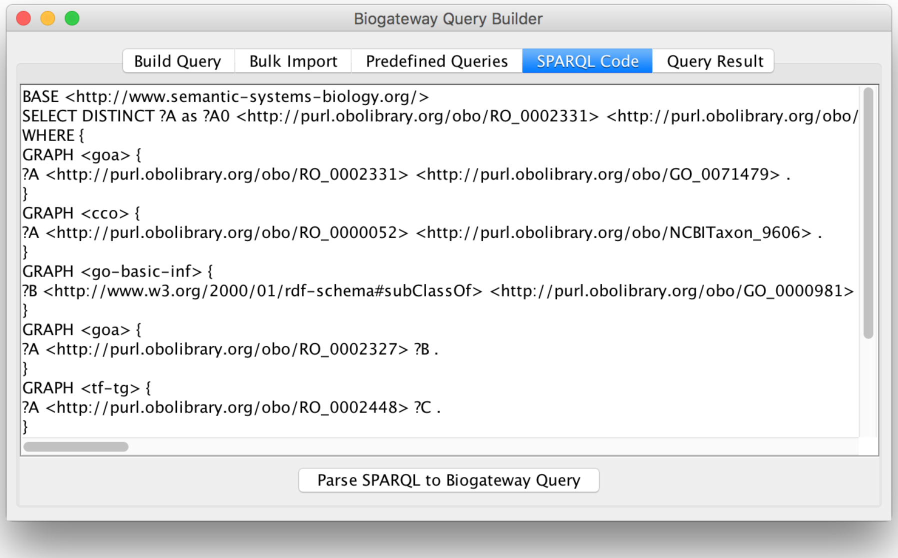
Only show relations to nodes in current network

This checkbox will toggle a filter that will only show the results that include a node that exists in the current network. This is useful for comparing query results with the network that the user is actively working with.

Import to new/selected Network

These buttons will import the selected relations to a new Cytoscape network, or the currently selected one. New networks created with BioGateway will automatically apply the BioGateway visual style.

5.2.5 SPARQL Code



```
BASE <http://www.semantic-systems-biology.org/>
SELECT DISTINCT ?A as ?A0 <http://purl.obolibrary.org/obo/RO_0002331> <http://purl.obolibrary.org/obo/
WHERE {
GRAPH <goa> {
?A <http://purl.obolibrary.org/obo/RO_0002331> <http://purl.obolibrary.org/obo/GO_0071479> .
}
GRAPH <cco> {
?A <http://purl.obolibrary.org/obo/RO_0000052> <http://purl.obolibrary.org/obo/NCBITaxon_9606> .
}
GRAPH <go-basic-inf> {
?B <http://www.w3.org/2000/01/rdf-schema#subClassOf> <http://purl.obolibrary.org/obo/GO_0000981>
}
GRAPH <goa> {
?A <http://purl.obolibrary.org/obo/RO_0002327> ?B .
}
GRAPH <tf-tg> {
?A <http://purl.obolibrary.org/obo/RO_0002448> ?C .
}
```

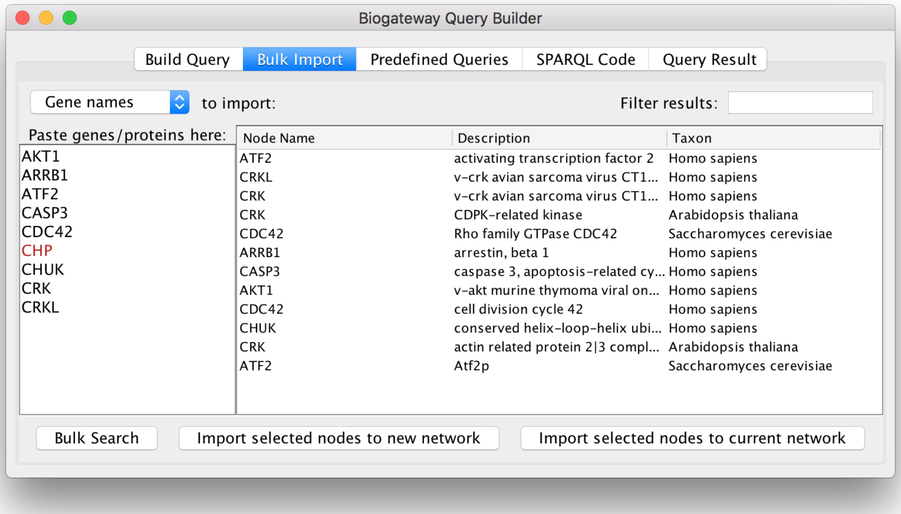
Figure 5.9: The SPARQL Code viewer

Clicking the button marked *"Generate SPARQL"* will show the *"SPARQL Code"* tab, which shows the SPARQL query generated by the current query in the Query Builder. This tab allows the user to modify the SPARQL code, and parse it back to the Query Builder. However, only minor changes are supported, as the SPARQL query must be structured in the exact same way for the Query Builder to represent it.

5.2.6 Saving and Loading queries

Queries built with the Query Builder can be saved and loaded as text files for later use. The files will be stored with the extension *.bgwsparql*, and stored in the same format as the SPARQL code shown in the *"SPARQL Code"* tab. The app will remember the last used location to save and load queries, so directories can be used to organise the stored queries.

5.3 Using the Bulk Importer



Gene names to import:

Filter results:

Paste genes/proteins here:

AKT1
ARRB1
ATF2
CASP3
CDC42
CHP
CHUK
CRK
CRKL

Node Name	Description	Taxon
ATF2	activating transcription factor 2	Homo sapiens
CRKL	v-crk avian sarcoma virus CT1...	Homo sapiens
CRK	v-crk avian sarcoma virus CT1...	Homo sapiens
CRK	CDPK-related kinase	Arabidopsis thaliana
CDC42	Rho family GTPase CDC42	Saccharomyces cerevisiae
ARRB1	arrestin, beta 1	Homo sapiens
CASP3	caspase 3, apoptosis-related cy...	Homo sapiens
AKT1	v-akt murine thymoma viral on...	Homo sapiens
CDC42	cell division cycle 42	Homo sapiens
CHUK	conserved helix-loop-helix ubi...	Homo sapiens
CRK	actin related protein 2|3 compl...	Arabidopsis thaliana
ATF2	Atf2p	Saccharomyces cerevisiae

Figure 5.10: Importing a list of genes with the Bulk Importer

The Bulk Importer is a part of the BioGateway Query Builder, and can be used to quickly import a set of genes, proteins or GO terms into Cytoscape in a BioGateway compatible network. The results can be filtered to match a specific name, description or taxon, and can be selected and imported into a new network, or the currently active one.

The nodes that were not found are marked in red after the query has been run.

5.3.1 Searching for gene/protein names

To search for genes or proteins by their names, select either *"Gene names"* or *"Protein names"* in the pull-down menu in the upper left corner. Paste a list of names to search for, with one name on each line. The tool will only search for exact matches of the names, for searching for partial matches, the *"Add BioGateway node"* feature described in section 5.4.1 must be used.

5.3.2 Searching for UniProt IDs

To search for a list of UniProt IDs, select *"UniProt IDs"* from the dropdown menu in the upper left corner. The tool will look for nodes matching the URI by appending the UniProt ID to *"http://identifiers.org/uniprot/"*. For example, when looking up the UniProt ID *"P04637"*, it will generate the URI *"http://identifiers.org/uniprot/P04637"*. The tool will then return these nodes if they are found.

5.3.3 Searching for GO terms

To search for a set of GO term IDs, select *"GO terms"* in the upper left corner. The terms must be on the form *"GO: ... "*. The tool will generate an URI from the GO term. As an example, the term *"GO:0003674"* will result in the URI *"http://purl.obolibrary.org/obo/GO_0003674"*. The tool will then return any nodes with the URIs generated.

5.4 Network right-click menus

Several functions of the BioGateway app are easily accessible by right-clicking in the network view of Cytoscape. Depending on whether none, one or several nodes are selected when right-clicking, different menus will be available. If several nodes are selected, it does not matter if the user right-clicks on the background or on a node. If no node is selected and the user right-clicks on a node, the action will be the same as if that node was selected.

Right-clicking on an edge enables a specific set of actions.

5.4.1 Adding new nodes to the network

If the user right-clicks on the network background (not on a node or edge) while no node is selected, they will be presented with the option to *"Add BioGateway node"*. This is a handy feature for quickly adding specific nodes to a network, and will open the Node Lookup view described in section 5.2.2.

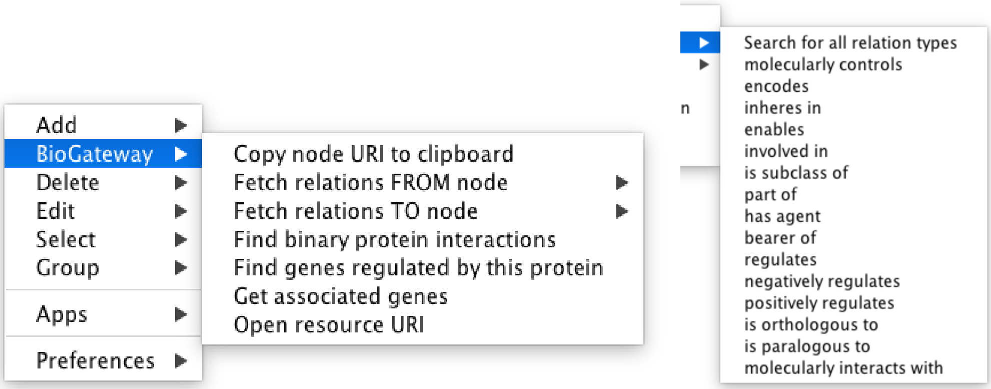
5.4.2 Relation source data

By right clicking on an edge and selecting "View source data", a window appears with additional information about the relation represented by the edge. For relations supported by PubMed IDs, these are available and can be opened in the default web browser for further review. The relation type definition can also be opened in the same way.

5.4.3 Single node queries

Copy node URI to clipboard

This function will simply copy the selected node URI to the clipboard. This is useful for using the URIs of nodes in an existing network to construct a query with the Query Builder tool.



(a) The available functions and queries for a single node.

(b) Searching for a specific relation to or from a node.

Relation Type	From Node type	To Node type	Dataset graph
Molecularly controls	Proteins	Genes	TF-TG
Encodes	Genes	Proteins	Refseq
Inheres in	Proteins or Genes	Taxa	Refseq and Refprot
Enables	Proteins	GO Terms	GOA
Inheres in	Proteins	GO Terms	GOA
Has agent	Protein-Protein Interactions	Proteins	IntAct

Table 5.1: Relation directions for the most commonly used relation types.

Fetch relations FROM/TO node

The queries *"Fetch relations FROM node"* and *"Fetch relations TO node"* are essentially the same, but searches in different directions. When searching *FROM*, the selected node is the node before the relation, and when searching *TO*, it is set as the node after the relation.

It is important to understand how the datasets are structured in order to know which direction the relations should be search for. Most datasets only have relations in one direction, going from one type of node to another type. See table 5.1 for an overview of relation directions in some of the most relevant datasets.

Find binary protein interactions

This menu action is only available if the selected node is a protein.

While *"binary PPIs"* is not a concept supported by the BioGateway server yet, this menu action will generate a query that will achieve the same effect. It will search for any PPI with only two participants where the source protein is one of them, and return a relation of the type *"molecularly interacts with"* pointing to the other participant.

When imported to a Cytoscape network the created edge will be labelled as bi-directional, and the BioGateway visual style will show it with arrows in both ends.

Find genes regulated by this protein

This action is a shortcut to searching for transcription factors or target genes. If the selected node is a gene, it will be shown as “Find proteins regulating this gene” instead.

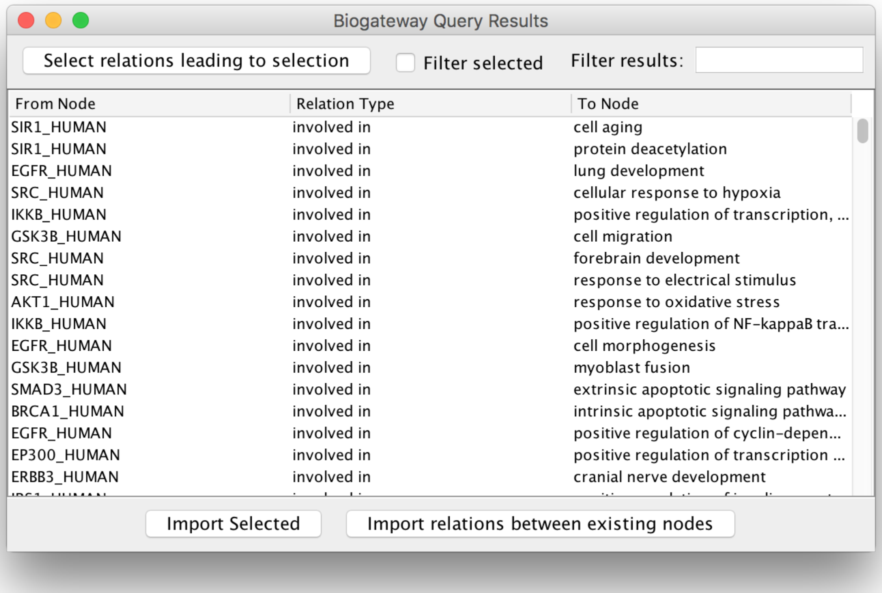
Get associated genes/proteins

This action is a shortcut to getting the genes encoding a protein, or proteins encoded by a gene, as these are common actions.

Open resource URI

This menu action will open the URI of the selected node as an URL in the default web browser for further details about the biological entity.

5.4.4 Importing results


Biogateway Query Results
Select relations leading to selection
□ Filter selected
Filter results:
Import Selected
Import relations between existing nodes

From Node	Relation Type	To Node
SIR1_HUMAN	involved in	cell aging
SIR1_HUMAN	involved in	protein deacetylation
EGFR_HUMAN	involved in	lung development
SRC_HUMAN	involved in	cellular response to hypoxia
IKKB_HUMAN	involved in	positive regulation of transcription, ...
GSK3B_HUMAN	involved in	cell migration
SRC_HUMAN	involved in	forebrain development
SRC_HUMAN	involved in	response to electrical stimulus
AKT1_HUMAN	involved in	response to oxidative stress
IKKB_HUMAN	involved in	positive regulation of NF-kappaB tra...
EGFR_HUMAN	involved in	cell morphogenesis
GSK3B_HUMAN	involved in	myoblast fusion
SMAD3_HUMAN	involved in	extrinsic apoptotic signaling pathway
BRCA1_HUMAN	involved in	intrinsic apoptotic signaling pathwa...
EGFR_HUMAN	involved in	positive regulation of cyclin-depen...
EP300_HUMAN	involved in	positive regulation of transcription ...
ERBB3_HUMAN	involved in	cranial nerve development

Figure 5.12: The Query Result window.

When executing a query from a network, the result view shown in figure 5.12 will be slightly different than the results in figure 5.2. As the network was initiated from a network’s view, the results will always be imported to the same network. The *"Import Selected"* button in will import the relations in the selected rows into the currently active network.

The *"Import relations between existing nodes"* button will import all relations found in the current query, which are between nodes already present in the current network. No new nodes will be added. Note that this does not require any relations to be selected, all relations in the result which are between two existing nodes will be added. This feature is useful for finding all relations of a current type in a network, especially if used in conjunction with multi-node queries.

5.4.5 Multi-node queries

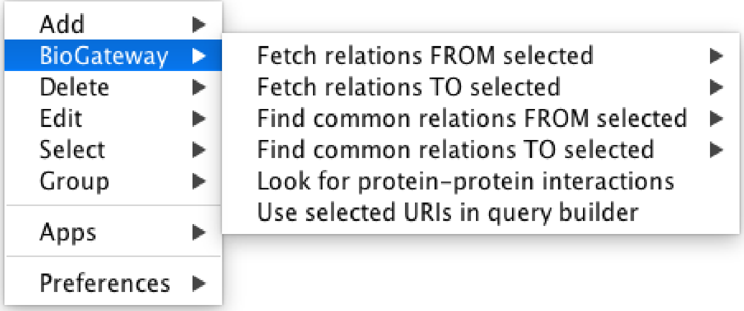

Add ▶
BioGateway ▶
Delete ▶
Edit ▶
Select ▶
Group ▶
Apps ▶
Preferences ▶
Fetch relations FROM selected ▶
Fetch relations TO selected ▶
Find common relations FROM selected ▶
Find common relations TO selected ▶
Look for protein–protein interactions
Use selected URIs in query builder

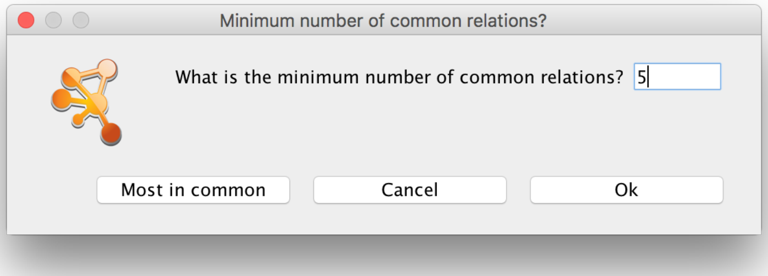
Figure 5.13: The right-click context menu when more than one node is selected.

Multi-node queries work similar to the single-node queries, but they are applied to all selected nodes. Such queries can be an efficient way to expand a network.

Fetch relations FROM/TO selected

This query works just like its single node counterparts. All the resulting relations will be shown together in the Query Result window after completion.

Find common relations FROM/TO selected


Minimum number of common relations?

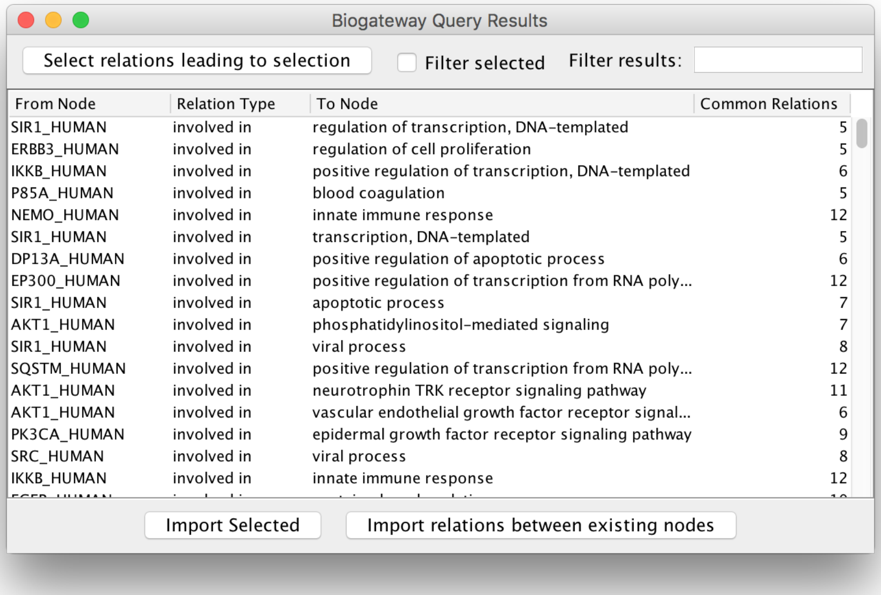
What is the minimum number of common relations? 5

Figure 5.14: Selecting the minimum number of common relations.

This feature lets the user search over a set of nodes and only get the relations pointing to new nodes which have several relations with the ones in the searched set. For instance, if the user want to search for GO terms that a set of proteins are involved in, but filter out the less common terms, the user could set a minimum threshold of common relations.

Example:

- The user is looking for GO terms that at least 5 of the proteins in the search set are involved in.
- The user would type in *"5"* as the minimum number of common relations (figure 5.14).
- Only the GO terms which at least 5 distinct proteins from the search set are involved in, are shown.
- The results can be sorted by number of common relations.


Biogateway Query Results
Select relations leading to selection
□ Filter selected
Filter results:
Import Selected
Import relations between existing nodes

From Node	Relation Type	To Node	Common Relations
SIR1_HUMAN	involved in	regulation of transcription, DNA-templated	5
ERBB3_HUMAN	involved in	regulation of cell proliferation	5
IKKB_HUMAN	involved in	positive regulation of transcription, DNA-templated	6
P85A_HUMAN	involved in	blood coagulation	5
NEMO_HUMAN	involved in	innate immune response	12
SIR1_HUMAN	involved in	transcription, DNA-templated	5
DP13A_HUMAN	involved in	positive regulation of apoptotic process	6
EP300_HUMAN	involved in	positive regulation of transcription from RNA poly...	12
SIR1_HUMAN	involved in	apoptotic process	7
AKT1_HUMAN	involved in	phosphatidylinositol-mediated signaling	7
SIR1_HUMAN	involved in	viral process	8
SQSTM_HUMAN	involved in	positive regulation of transcription from RNA poly...	12
AKT1_HUMAN	involved in	neurotrophin TRK receptor signaling pathway	11
AKT1_HUMAN	involved in	vascular endothelial growth factor receptor signal...	6
PK3CA_HUMAN	involved in	epidermal growth factor receptor signaling pathway	9
SRC_HUMAN	involved in	viral process	8
IKKB_HUMAN	involved in	innate immune response	12

Figure 5.15: Results from a common relations search.

Look for binary PPIs

This query works in the same way as its corresponding single-node query, and will be executed for all the currently selected nodes. This option will only appear if at least one protein is among the selected nodes.

Look for common binary PPIs

This works the same way as finding common relations to/from selected nodes, described above, but searches for binary PPIs instead, as described in section 5.4.3.

Use selected URIs in query builder

This feature will open a new Query Builder window with the URIs of the selected nodes already filled out on the left side. By using the swap button, the URIs can be placed on the desired side of the relation if needed.

5.5 Understanding the Datasets

BioGateway contains several graphs from many different sources. This is a short overview of some of the most relevant graphs, and which datasets they are imported from.

5.5.1 Gene Ontology

GO-BASIC

This dataset contains the GO terms from the Gene Ontology Consortium[26]. The most relevant relations in this graph are:

- *"subclass of"* - From GO terms to their parent GO terms.
- *"part of"* - A GO term can be involved in, or part of, another GO term.
- *"regulates"* - Regulations between GO terms.
- *"positively regulates"* - From GO terms to GO terms they positively regulate.
- *"negatively regulates"* - From GO terms to GO terms they negatively regulate.

5.5.2 Gene Ontology Annotations

The GOA dataset contains GO term annotations of proteins. The relations included are:

- *“involved in”* - for biological processes
- *“part of”* - for cellular components
- *“enables”* - for molecular functions

5.5.3 RefProt

This is a graph containing protein data from the UniProt Reference Proteomes[66]. It includes the relation types:

- *“inheres in”* - A relation from a protein to the organism (taxon) it belongs to.
- *“involved in”* - Relations from proteins to diseases they are involved in.
- *“bearer of”* - Relations from proteins to protein modifications.
- *“encodes”* - Relations from a protein to the gene(s) that encode it.

5.5.4 RefSeq

This graph contains data from NCBIGene's RefSeqGene[51] dataset. The relation types included are:

- *"encodes"* - Relations from genes to the proteins they encode for.
- *"inheres in"* - Relations from genes to the taxa they inhere in.

5.5.5 IntAct

The IntAct[12] dataset contains Protein-Protein Interaction (PPI) data. The most important relation in this dataset is the *"has agent"* relation. The graph does not consist of relations between proteins, but rather considers the PPIs as nodes themselves, and thus consists of the PPIs and their participating proteins, annotated as agents. While some PPIs are binary, and only have two participants, others have multiple participating proteins.

This dataset can be used to search for proteins participating in the same PPIs. The BioGateway app also has a helper function to find binary PPIs directly. In this case, the app will hide the PPI nodes and infer the *"molecularly interacts with"* relations between the two proteins. This only applies to PPIs with exactly two participants, to find proteins participating in non-binary PPIs, first search for PPIs with the *"has agent"* relation to the protein of interest, and then search for *"has agent"* relations from these PPIs. This approach should also be used when searching for PPIs in the Query Builder.

5.5.6 TF-TG

This dataset of transcription factors and target genes (TG-TF) is not currently published, but is available in BioGateway through a collaboration with the authors as a beta dataset. It contains almost 190 000 transcription factor annotations. This dataset should be used for testing purposes only until it is published.

5.6 The BioGateway Visual Style

The BioGateway app comes with its own visual style to highlight the different types of nodes and relations available through BioGateway. When creating new networks through the Query Builder, it will automatically be applied. If the user does not have any Cytoscape style named *"BioGateway"*, it will create a new one. The user is free to modify the style to their own liking, and modifications will not be overwritten when new networks are created, as long as the style is not deleted. To revert to the default BioGateway style, simply delete it from the Cytoscape visual style manager, and create a new network through the Query Builder.

CHAPTER 6

Conclusion

6.1 Increased accessibility

As stated in section 2.6, the goal of this thesis was to develop a more intuitive user interface and visualisation tool for interacting with the BioGateway database. By building on top of the popular Cytoscape platform, we have created a plugin - or app, as it is called - that enables users to access the powerful query features of BioGateway from within Cytoscape. The feedback from test users given in chapter 4.2 has been clear in their opinion that this app provides an accessible interface to BioGateway, and in the use cases where BioGateway has all the datasets needed to build a network, it can provide great improvements in the workflow compared to copying and parsing data to and from web resources. There are still some user interface issues that can be improved and features which can be added, but it seems clear that this app has demonstrated that the underlying architecture and datasets of BioGateway still is relevant today, and that BioGateway has great potential if it is further developed and expanded with the commonly used datasets that the users have requested.

6.1.1 Too much of a good thing

Making the app too accessible, however, might not be desirable. If too much of the underlying complexities are hidden from the user, these nuances might be forgotten and result in incorrect results because the user is led to believe they are searching for something else than the app actually is looking for. It has therefore been a conscious decision to not oversimplify the use of relations in the BioGateway app. For instance, users have given feedback about the separation of genes and proteins into separate entities, instead of unifying them, because they are “always the same”, but in some edge cases they are not. While having the extra set of nodes in a network might make them a little bit more complicated for the users, it is an important distinction to make for the sake of the few exceptions. Also, staying true to the data model of the BioGateway server makes the app simpler to use and more accurate in its reported results in the cases where such distinctions are important.

6.2 Further Work

While the master's thesis is complete when this document is submitted, the BioGateway project and its new Cytoscape app is not complete yet, nor should it be abandoned. The feedback from test users who want this tool to be improved and released, and from the researchers who are eager to see the BioGateway database include more datasets relevant to their research, has made it clear for us that we need to see this app through. Some tasks therefore remain before this project can be considered complete.

6.2.1 More data sources in BioGateway

In the feedback from users, it was clear that more datasets were desired. The test users has suggested that BioGateway needs support for:

- Signaling networks (Omnipath, SIGNOR)
- miRNAs-target interactions
- Metabolic network datasets (BIGG Models)
- Drug/small compound to gene interactions (DrugBank)
- Drug/small compound to PPIs.

MLA has also been kind enough to provide a curated subset of OmniPath for inclusion to BioGateway. This dataset would provide protein to TF regulation data, and is considered for inclusion into BioGateway in the near future.

6.2.2 Improved UI features

As new funcionality has been implemented, requests for new features has been coming in as well. While the user interface of the current version is good enough to be useful in many situations, some features had to be put off to a later version.

Most notably is how the relations in the Query Builder will all need to be true for results to be found. Allowing for relations to be grouped with each other and joined by logical *AND* or *OR*, will enable much richer expressions and queries. This will require a significant rework of the Query Builder user interface, and the underlying models that create, parse, load and save queries, but could be an undertaking worth doing given the added flexibilities of the tool.

6.2.3 Add more predefined queries to the config XML

While the functionality of the Query Builder and querying from selected nodes are powerful features, there are some types of queries that cannot be formulated in the Query Builder. While the "Predefined Queries" functionality has not been given much attention in this thesis, it can be useful in specific use cases where the support for such queries has been added in the configuration XML file. As long as the desired queries can be built in a reasonable way in SPARQL, they can quickly be added to the remote configuration file, and be available to end users the next time they open Cytoscape.

6.2.4 Open-sourcing the project

The BioGateway app source code is already publicly available at Bitbucket[67], but some work remain to make it easier for others to contribute and maintain the project. Old, unused code should be removed from the codebase, and the source code should be commented well enough for others to understand the data models and data flows in the application.

As the project is moving out of Beta stage, the project itself should be set up to create release versions which should be downloadable from the repository, and the repository's main page should include a manual explaining how to install and use the BioGateway app.

6.2.5 Release the app in the Cytoscape App Store

To reach more potential users and make installation and updating easier, the app should be submitted and published through the Cytoscape App Store[42]. While submitting the app in itself is simple, it needs to be thoroughly tested to assure that it will not interfere with other apps, or crash Cytoscape itself.

6.2.6 Publish a paper about the new App

The BioGateway project has previously published articles about its functionalities and possibilities, and the ability to access these features through a Cytoscape app would ideally be disseminated through the biology and bioinformatics communities. Given the response from the test users involved in this project, we believe this tool will be of value to a sizeable user community.

6.2.7 Updating the BioGateway server to increase performance

While the Virtuoso server being used now is performing reasonably well, there are some issues with performance in certain situations, among them an issue which prevents us from loading many names and descriptions of the query results in the initial query. For the queries that returns hundreds, or thousands of nodes, this means that each name and description needs to be loaded in independent queries, causing unnecessarily long load times.

Such slowdowns should be resolved before the app is published and used by hundreds, if not thousands of users following an app release and publication, otherwise the server might get overloaded, user experience will suffer, and BioGateway's reputation with it. The maintainers of the BioGateway servers are looking into newer, more robust architectures for running the SPARQL server.

6.2.8 Support for importing other networks

As discussed in section 4.2.3, a helper tool for converting networks from other sources into a BioGateway-compatible network with the "identifier uri" field for all nodes would be a great addition to the app. Also, for the apps that are open-source, adding support for this field as a suggested update to their source code could be a way to make them compatible with apps depending on URIs such as BioGateway.

6.3 Reflection

With my background from computer science, I was quite shocked when I came to learn that even state-of-the-art biomedical research communities still relied on the spreadsheet as their daily database solution, and how modern datasets still seem to be influenced by this (like column 16). So to me, the advantages of Semantic Systems Biology seemed obvious, but it remained hard to convince the average biologists when they could not see it for themselves.

While working with RR and MLA I got insights into how they saw things, and how we saw things differently. With my programmer background I think of data and information in a structured and object-oriented way, with clearly defined relationships and protocols between them, while they showed me how they would be working with data as rows in a spreadsheet. This works very well on small scales, but can quickly become unmanageable, and I think there is a dire need for better data modeling tools for the Life Sciences that are better equipped to handle all the different kinds of data added to a dataset through its lifetime. I believe Semantic Web technologies might be a good step forward in this regard, but in the end the users need easy tools that make their everyday workflow easier, and BioGateway has the potential to become one such tool.

Acknowledgements

First I would like to thank my supervisor from the Department of Biology at NTNU, Prof. Martin Kuiper, for all his patience and efforts in helping me make sense of the field of biology and bioinformatics, of which I had very little expertise in from before. If my lack of a biology background is not apparent in this thesis, it is because of his guidance and teachings, for which I am very thankful. I feel that Martin has not only treated me like his student, but also like a colleague and friend, and he has been very kind to involve me in his academic group.

I would also like to thank Vladimir Mironov for our many insightful discussions about the inner workings of everything from database systems to coffee grinders. His knowledge and passion for maintaining and improving BioGateway has been absolutely instrumental in making this app work as well as it does, and I feel that I have been given credit for many features which has been built on the powerful features made possible by Vladimir. He made great efforts to integrate the unpublished TF-TG data so that it could be used in this app, without it many of the features shown would not be possible.

Rafael “Rafa” Riudavets has been a great office mate, friend, and test user. His feedback has proved invaluable in designing and bug fixing this tool, and his suggestions has improved the functionality a lot. He has always taken the time to answer my questions, and has been indispensable in translating biological concepts to a software engineer like me.

Marcio Luis Acencio has been very engaged and positive about my project, and has always been looking for new opportunities for the app. Our discussions have been very enlightening and fruitful, despite our different academic backgrounds, and his enthusiasm for providing new datasets to BioGateway has been very inspiring.

Finally, I would like to thank my primary supervisor, Prof. Keith Downing of the Department of Computer Science, for his patience and belief in me. He has always been there to push me in the right direction when needed. His contagious optimism has been most appreciated in the last frantic weeks of writing, and kept convincing me that everything would be fine in the end.

Bibliography

[1] Hiroaki Kitano. “Systems Biology: A Brief Overview”. In: *Science* 295.5560 (2002), pp. 1662–1664. ISSN: 0036-8075.

[2] Trey Ideker et al. “Integrated Genomic and Proteomic Analyses of a Systematically Perturbed Metabolic Network”. In: *Science* 292.5518 (2001), pp. 929–934. ISSN: 0036-8075.

[3] IBM. *Blue Gene L.* URL: `http://www-03.ibm.com/press/us/en/photo/10146.wss` (visited on 11/30/2017).

[4] International Human Genome Sequencing Consortium. “Initial sequencing and analysis of the human genome”. In: *Nature* 409 (Feb. 2001).

[5] Mark Schena et al. “Quantitative Monitoring of Gene Expression Patterns with a Complementary DNA Microarray”. In: *Science* 270.5235 (1995), pp. 467–470. ISSN: 0036-8075.

[6] F. Allen et al. “Blue gene: A vision for protein science using a petaflop supercomputer”. English. In: *IBM Systems Journal* 40.2 (2001), pp. 310–327.

[7] IBM. *IBM Research Journal 49, 2005.* URL: `http://www.research.ibm.com/journal/rd49-23.html` (visited on 11/30/2017).

[8] Michael Shirts and Vijay S. Pande. “Screen Savers of the World Unite!” In: *Science* 290.5498 (2000), pp. 1903–1904. ISSN: 0036-8075.

[9] Michael Gross. “Folding research recruits unconventional help”. In: *Current Biology* 22.2 (2012), R35–R38. ISSN: 0960-9822.

[10] Frank J. Bruggeman and Hans V. Westerhoff. “The nature of systems biology”. In: *Trends in Microbiology* 15.1 (2007), pp. 45–50. ISSN: 0966-842X.

[11] UniProt Consortium. *UniProt.* URL: `http://www.uniprot.org` (visited on 11/30/2017).

[12] EMBL-EBI. *IntAct.* URL: `https://www.ebi.ac.uk/intact/` (visited on 11/30/2017).

[13] National Center for Biotechnology Information. *NCBI Gene.* URL: `https://www.ncbi.nlm.nih.gov/gene` (visited on 11/30/2017).

[14] Bronwen L. Aken et al. “The Ensembl gene annotation system”. In: *Database* 2016 (2016), baw093.

[15] World Wide Web Consortium. *Semantic Web.* URL: `http://www.w3.org/standards/semanticweb/` (visited on 11/30/2017).

[16] Christian Bizer, Tom Heath, and Tim Berners-Lee. “Linked Data - the story so far”. 2009.

[17] World Wide Web Consortium. *RDF Specification.* URL: https://www.w3.org/RDF/ (visited on 11/30/2017).

[18] World Wide Web Consortium. *Semantic Web.* URL: http://www.w3.org/TR/sparql11-overview/ (visited on 11/30/2017).

[19] Gary Bader. *Pathguide.* URL: http://www.pathguide.org (visited on 11/30/2017).

[20] Xiaoshu Wang, Robert Gorlitsky, and Jonas S Almeida. "From XML to RDF: how semantic web technologies will change the design of 'omic' standards". In: *Nature Biotechnology* 23 (Sept. 2005).

[21] Alan Ruttenberg et al. "Advancing translational research with the Semantic Web". In: *BMC Bioinformatics* 8.3 (Nov. 2007), S2. ISSN: 1471-2105.

[22] World Wide Web Consortium. *Semantic Web Health Care and Life Sciences (HCLS) Interest Group.* URL: https://www.w3.org/2001/sw/hcls/ (visited on 11/30/2017).

[23] Erick Antezana et al. "BioGateway: A semantic systems biology tool for the life sciences". In: *BMC Bioinformatics* 10.10 (Oct. 2009), S11. ISSN: 1471-2105.

[24] Erick Antezana, Martin Kuiper, and Vladimir Mironov. "Biological knowledge management: the emerging role of the Semantic Web technologies". In: *Briefings in Bioinformatics* 10.4 (2009), pp. 392–407.

[25] Erick Antezana, Vladimir Mironov, and Martin Kuiper. "The emergence of Semantic Systems Biology". In: *New Biotechnology* 30.3 (2013). Functional Genomics Reviews, pp. 286–290. ISSN: 1871-6784.

[26] Gene Ontology Consortium. *Gene Ontology.* URL: http://www.geneontology.org (visited on 11/30/2017).

[27] Donna Maglott et al. "Entrez Gene: gene-centered information at NCBI". In: *Nucleic Acids Research* 33.Database Issue (Jan. 2005), pp. D54–D58.

[28] Kanehisa Labs. *Kyoto Encyclopedia of Genes and Genomes.* URL: http://www.kegg.jp (visited on 11/30/2017).

[29] FranÃ§ois Belleau et al. "Bio2RDF: Towards a mashup to build bioinformatics knowledge systems". In: *Journal of Biomedical Informatics* 41.5 (2008). Semantic Mashup of Biomedical Data, pp. 706–716. ISSN: 1532-0464.

[30] Bio2RDF. *Bio2RDF SPARQL Endpoint.* URL: http://bio2rdf.org/sparql (visited on 11/30/2017).

[31] NTNU. *Martin Kuiper.* URL: https://www.ntnu.edu/employees/martin.kuiper (visited on 12/01/2017).

[32] Gene Ontology Consortium. *AmiGO 2.* URL: http://amigo.geneontology.org/amigo (visited on 11/30/2017).

[33] UniProt Consortium. *TP53 - Cellular tumor antigen p53.* URL: http://www.uniprot.org/uniprot/P04637 (visited on 11/30/2017).

[34] InterPro EMBL-EBI. *InterPro protein sequence analysis & classification.* URL: https://www.ebi.ac.uk/interpro/ (visited on 12/01/2017).

[35] UniProt Consortium. *UniProt SPARQL Endpoint.* URL: http://sparql.uniprot.org (visited on 11/30/2017).

[36] Bio2RDF. *Bio2RDF Home.* URL: https://github.com/bio2rdf/bio2rdf-scripts/wiki (visited on 11/30/2017).

[37] Bio2RDF. *About: 5-hydroxytryptamine (serotonin) receptor 1B.* URL: http://bio2rdf.org/ncbigene:37191 (visited on 11/30/2017).

[38] Cytoscape Consortium. *Cytoscape Home Page.* URL: http://www.cytoscape.org/ (visited on 11/30/2017).

[39] yWorks. *Major Layout Algorithms.* URL: https://docs.yworks.com/yfiles/doc/api-json/#/dguide/major_layouters (visited on 11/30/2017).

[40] Cytoscape Consortium. *Visual Styles.* URL: http://wiki.cytoscape.org/Cytoscape_User_Manual/Visual_Styles (visited on 11/30/2017).

[41] P. Shannon et al. "Cytoscape: a software environment for integrated models of biomolecular interaction networks". In: *Genome Res.* 13.11 (Nov. 2003), pp. 2498–2504.

[42] Cytoscape Consortium. *Cytoscape App Store.* URL: http://apps.cytoscape.org/ (visited on 11/30/2017).

[43] Cytoscape Consortium. *Cytoscape App Cookbook.* URL: http://wiki.cytoscape.org/Cytoscape_3/AppDeveloper/Cytoscape_3_App_Cookbook (visited on 11/30/2017).

[44] Andrea Splendiani. "RDFScape: Semantic Web meets Systems Biology". In: *BMC Bioinformatics* 9.Suppl 4 (2008), S6–S6.

[45] Andrea Splendiani. *RDFScape Home.* URL: http://bioinformatics.org/rdfscape/wiki/ (visited on 11/30/2017).

[46] General Bioinformatics. *General Bioinformatics - About us.* URL: https://www.generalbioinformatics.com/aboutus.html (visited on 11/30/2017).

[47] General Bioinformatics. *General SPARQL Home.* URL: http://generalbioinformatics.github.io/general-sparql-cy3/ (visited on 11/30/2017).

[48] GitHub. *General SPARQL GitHub Repository.* URL: https://github.com/generalbioinformatics/general-sparql-cy3 (visited on 11/30/2017).

[49] OBO Technical WG. *The OBO Foundry.* URL: http://www.obofoundry.org/ (visited on 11/30/2017).

[50] National Center for Biotechnology Information. *Taxonomy - NCBI.* URL: https://www.ncbi.nlm.nih.gov/taxonomy (visited on 11/30/2017).

[51] National Center for Biotechnology Information. *RefSeqGene.* URL: https://www.ncbi.nlm.nih.gov/refseq/rsg/ (visited on 11/30/2017).

[52] PAZAR Group. *PAZAR.* URL: http://www.pazar.info/ (visited on 11/30/2017).

[53] de Duve Institute. *TFactS.* URL: http://www.tfacts.org (visited on 11/30/2017).

[54] Network Biology Laboratory. *TRRUST - Transcriptional Regulatory Relationships Unraveled by Sentence-based Text mining.* URL: http://www.grnpedia.org/trrust/ (visited on 11/30/2017).

[55] OpenLink. *OpenLink Virtuoso.* URL: https://virtuoso.openlinksw.com/ (visited on 11/30/2017).

[56] BioGateway. *Querying BGW and APOs with SPARQL.* URL: `http://www.semantic-systems-biology.org/querying` (visited on 11/30/2017).

[57] Antonio Fabregat et al. “The Reactome Pathway Knowledgebase”. In: *Nucleic Acids Research* (2017), gkx1132.

[58] Huaiyu Mi et al. “PANTHER version 11: expanded annotation data from Gene Ontology and Reactome pathways, and data analysis tool enhancements”. In: *Nucleic Acids Research* 45.D1 (2017), pp. D183–D189.

[59] Livia Perfetto et al. “SIGNOR: a database of causal relationships between biological entities”. In: *Nucleic Acids Research* 44.D1 (2016), pp. D548–D554.

[60] Ethan G. Cerami et al. “Pathway Commons, a web resource for biological pathway data”. In: *Nucleic Acids Research* 39.suppl_1 (2011), pp. D685–D690.

[61] Glyn Bradley and Steven J Barrett. “CausalR: extracting mechanistic sense from genome scale data”. In: *Bioinformatics* 33.22 (), pp. 3670–3672. (Visited on 11/30/2017).

[62] Luiz A. Bovolenta, Marcio L. Acencio, and Ney Lemke. “HTRIdb: an open-access database for experimentally verified human transcriptional regulation interactions”. In: *BMC Genomics* 13.1 (Aug. 2012), p. 405. ISSN: 1471-2164.

[63] Steven Maere, Karel Heymans, and Martin Kuiper. “BiNGO: a Cytoscape plugin to assess overrepresentation of Gene Ontology categories in Biological Networks”. In: *Bioinformatics* 21.16 (2005), pp. 3448–3449.

[64] Kris Laukens et al. “Flexible network reconstruction from relational databases with Cytoscape and CytoSQL”. In: *BMC Bioinformatics* 11.1 (July 2010), p. 360. ISSN: 1471-2105.

[65] NTNU. *Systems Biology: Resources, standards and tools.* URL: `https://www.ntnu.no/studier/emner/BI3019` (visited on 11/30/2017).

[66] UniProt Consortium. *UniProt Reference Proteomes.* URL: `http://www.uniprot.org/help/reference_proteome` (visited on 11/30/2017).

[67] Bitbucket. *BioGateway Cytoscape Plugin Repository.* URL: `https://bitbucket.org/druglogics/biogw_cytoscape` (visited on 12/01/2017).

Appendices

APPENDIX A

Test User Feedback

These are the unedited feedback reports received from use case providers and app testers, Rafael Riudavets and Marcio Luis Acencio. Many of their comments were translated into current app functionality, but a lot still remains for future work: some of which will need to be completed before the release version will be submitted for publication.

A.1 Feedback from Marcio Luis Acencio

A.1.1 What is your current line of work?

Currently I am working as a biocurator involved in the conversion of unstructured knowledge of transcription regulatory activity of sequence-specific DNA binding transcription factors (DbTFs) into structured and computable knowledge. This conversion involves the extraction of knowledge from biomedical literature, the creation of curation guidelines to secure a precise conversion and the development of appropriate ontology terms to describe the acquired knowledge.

A.1.2 Which resources do you normally use when construction networks?

Currently I have not worked on construction of networks, but in my previous projects I had to build many different networks. My networks (called integrated networks because they contain simultaneously protein-protein, metabolic and TF-TG interactions) were constructed mainly by parsing the tab-limited files downloaded from diverse resources, such as BioGRID, YEASTRACT, DIP, HPRD, IntAct, MINT, MIPS, MPPI, HIPPIE, TRED and metabolic networks from the resource "BIGG Models". But I have also built my own self-curated dataset - the Human Transcriptional Regulation Interactions Database (HTRIdb).

A.1.3 How do you use these resources?

Currently I haven't used any of the above-mentioned resources. But I can describe here my previous experience in constructing networks. My main mission in my previous projects was to construct integrated networks (containing simultaneously protein-protein, metabolic and TF-TG interactions) to use the topological measures as attributes in machine learning approaches to predict different types of biological phenomena (please check my NTNU's webpage for more details).

Therefore, I only used the resources for obtaining the data that I needed to construct my networks. But of course I had also to explore both the contents and the structure of these resources to select the interactions that I was interested in, namely only experimentally-evidenced interactions in which both interactors were known and from the same species. It is worth to mention that I had to parse the downloaded tab-limited files to filter out all interactions that were not appropriate for me. In addition, I had to parse these files to remove redundancy and to extract only the columns of my interest, namely the ones containing some identification for the interactors.

A.1.4 What were your first impressions of the BioGateway Cytoscape App?

My first impressions of BGW were very positive because I could detect a feature that it is very useful and virtually absent from other apps or resources: the quick and user-friendly construction of context-specific networks.

A.1.5 In what use cases does the app simplify your workflow?

I can provide one example of use case for which BGW could simplify my workflow. Such example is related to my current work as biocurator. Currently I have prepared a manuscript about the conversion of unstructured into computable knowledge of transcriptional regulatory activities of DbTFs. To try to show in the manuscript that such conversion is worthwhile, I have come up with some examples based on manual reasoning over Gene Ontology. In these examples I sought to demonstrate how a researcher can benefit from computable knowledge in comparison with a "traditional" biomedical literature survey. This manual reasoning approach was immensely simplified by BGW as shown below.

Use case: Why the expression of DbTF MafG is induced by cigarette smoke condensate?

1. Go to the biomedical literature.
    (a) Depends on just the keywords attached to the literature. Not semantically structured.
    (b) The literature is unstructured, so it's hard to find what you're looking for.
2. Using GOA acquire prior knowledge about the biological processes, molecular functions or cell parts the entities are involved in.
    (a) AmiGO or QuickGO
    (b) Looking at the GO terms the protein is associated with, trying to find terms you know to be associated with tumor cell viability, colony formation and invasion.

   (c) Found out that the protein is a known TF.

3. Look up the target genes of this TF protein.

   (a) Use a resource describing TFs, like TFactS, and target genes.

   (b) Using the target genes and their encoded proteins, return to step b. to find the GO terms these proteins are involved in.

4. Repeat 2 and 3 until results are found or no new data is available.

This is a manual and labor-intensive process, and involves translating data from one resource to another, usually by using text files and copy-pasting the data into web interfaces.

A.1.6 What does not work well in the BioGateway App?

1. The search for proteins in the URI lookup is not working well because it is not possible to find a protein by using the symbol or name of the corresponding encoding gene. For example, if I type "TP53" in the search field, I can't find the human version of this famous protein because the term "TP53" is associated only with the encoding gene. This should be more flexible.

2. Still regarding protein search: the way that search result appears is sometimes confusing because the terms shown in the column "Description" seem confusing. For example: if I type "TP53", the top protein found is "Q8QZZ7". Its description is "TP53RK-binding protein", but I think that it should be "TPRKB_MOUSE". For other proteins, the "Description" shows other types of terms and so on. So, it is not possible to guess what is really the Description. It would be more useful to have "Entry Name", "Protein names" and "Gene names" as columns in addition to "Node URI" and dsm

3. The results for searching PPIs are not intuitive. In fact researchers are not interested in the interactions identifiers, but, instead, in the interacting partners. This should be changed.

A.1.7 Which datasets do you feel BioGateway is missing?

I believe that BGW is missing (1) signaling networks-related datasets such as Omnipath or SIGNOR, (2) datasets containing miRNAs-targets interactions, (3) metabolic networks-related datasets such as BIGG Models, (4) datasets containing interactions between drug/small compound and gene products such as DrugBank and (5) datasets containing interactions between drug/small compound and protein-protein interactions.

A.1.8 Suggestions for new features or changes?

Some suggestions:

1. Allow selection of all types of interactions based on their confidence level

2. Allow selection of the relationships between GO terms and gene products based on the evidence codes used to support the annotation.

3. Include automated reasoning as new feature.

A.2 Feedback from Rafael Riudavets

A.2.1 What is your current line of work?

Protein activity prediction based on the integration of omics data.

A.2.2 Which resources do you normally use when construction networks?

To gather information I usually access databases such as KEGG, Reactome, IntAct, Panther Database, Signor, Pathway Commons, etc. To visualise the information I use Cytoscape.

A.2.3 How do you use these resources?

In my Bachelor Thesis I needed to build a network containing Protein-Protein Interactions (PPIs) and Transcription Factor - Target Genes (TF-TG) from different cancer related signaling pathways that were being targeted with drugs. Once we had the network, we would try to infer the activity state of the proteins that were part of our network based on the transcription profiles observed in every condition.

The process of building the whole network was divided into three steps: 1) building of the PPI network, 2) building the TF-TG network, 3) merging both networks to obtain a mixed network. To build the PPI network, information was extracted from KEGG (Release 81.0, January 1, 2017), where we focused on the networks that contained the nodes of interest, for which we would infer their activity state. Given that the software we wanted to use only accepted .sif (Simple Interaction Format) files, I manually converted the networks to this format. A total of 21 Transcription Factors (TFs) were included in the resulting network. To build the TF-TG network I extracted transcriptional regulatory relationships between the TFs present in our PPI network and their target genes. This information was taken from TFactS and TRRUST. Again, the information had to be converted to .sif format. Whenever I found an ambiguous interaction between two nodes I decided to remove both interactions, since I could not confirm which one was the correct one. Finally, I only needed to merge both networks to obtain a final network. The whole process took several hours, since I needed to find and download reliable resources, transform the data, and look for ambiguities.

After trying the Biogateway plugin for Cytoscape I could see that the same process could have been done in a much shorter time. For example, a fairly complex regulatory network (1370 nodes) could be built in one hour. Furthermore, the tool was also providing for every interaction a reference (PubMed ID) that I could check if I wanted to access the paper from where such interaction was extracted. This could also have been done with the original network, but it would have required to import tables into the built network. I saw how this tool not only provided access to a high-quality curated data, but also how to parse it very intuitively and in an efficient way to build the topology I needed for my study. Nevertheless, I also really enjoyed the flexibility of being able to select whether I wanted to expand the network in a certain direction or not, as well as being able to give direction to the regulation of an interaction based on different studies.

A.2.4 What were your first impressions of the BioGateway Cytoscape App?

It was really useful to be able to access the information so easily from Cytoscape. Instead of downloading a chunk of nodes and edges, I am able to expand the network step by step, which allows me to expand towards the direction I am interested in. The other apps let you download a predefined network, and afterwards you have to filter it, which may be virtually impossible if the created network is too big.

A.2.5 In what use cases does the app simplify your workflow?

In the process of building a network from a resource. This app integrates the process of looking for information and building a network in a single step. The extra feature that this app provides is that I am able to interactively expand from one or numeral nodes to a network. This is different from other apps. A lot of apps just give you an already built network and then you have to filter it to make it possible to work with. If I had to build the same network I explained before, the use of this App would have probably eased the process significantly, since I would have only needed to use one resource. To build the network, I would start by importing the genes I was interested in. Those genes are the ones that are being targeted by the different treatments used in the experiments (PI3K, MAP2K1/2, MAPK14). Once I had imported those nodes, I would have started expanding from there. To do so, I would have looked for proteins that are interacting with the initial proteins I just imported. This could be done by the right-clicking menu and selecting “Find binary protein interactions”. After getting the results, I would have parsed through them to try to find the ones that were involved in the pathway I wanted to study. Most importantly, I would look for transcription factors connected to the original proteins, since the objective of my thesis was to use those transcription factors as proxies of the activity state of the original proteins. To find the genes regulated by the transcription factors I just would have needed to select all the proteins, right-click and select “Find common relations FROM selected” > “TF-TG: molecularly controls”. Following this workflow I would have been able to create a network in a much more intuitive, easy and quick way than what I actually did. One important thing I found is that, at least until now, there is no information about metabolites in the database. This was relevant to me because one of the original proteins, PI3K, created a metabolite that was the one propagating the signal downstream, PIP3. This means that there is not a physical interaction between PI3K and the next protein in the signaling pathway, which means I will not find a connection between them in the database.

List format, just in case the explanation is too heavy:

1. Import original proteins (PI3K, MAP2K1/2, MAPK14): this are the proteins that are being targeted by the drugs used in the experiments.
2. Expand from the imported nodes: this can be done by right-clicking on the nodes and select “Find binary protein interactions”. After getting the results, I would parse through them to try to find the ones that are involved in the pathway I want to study. Most importantly, I would look for transcription factors connected to the original proteins, since the objective of my thesis was to use those transcription factors as proxies of the activity state of the original proteins.
3. Find the genes regulated by the TFs in the built network: this can be done by selecting all the proteins, right-clicking and select “Find common relations FROM selected” > “TF-TG: molecularly controls”.

A.2.6 Suggestions for new features or changes?

Maybe it would be good to have a more efficient way of setting the direction of regulation of entities. Since the resource prefers not to say that A activates/inhibits B because it may depend on the context, it would be good to have a way to complement the information gotten from Biogateway from the data retrieved from other databases. (For example being able to merge the created network with a SIF file in a way that it would update the direction of regulation between entities). I say that because one of the biggest problems I have encountered is that sometimes it is really tedious to integrate information from different resources in Cytoscape.